Vaadin 7

Joonas Lehtinen, PhD
Vaadin Ltd - CEO
vaadin.com/vaadin

@joonaslehtinen

Quick intro to

Vaadin Framework

vaadin }>

Vaadin is a UI framework for rich web applications

vaadin }>

java } html >

vaadin }>

goals

Developer experience

User expecience

Scalability

vaadin }>

102 ideas

vaadin }>

powerful components

vaadin }>

User Inteface

Data Source

Theme

vaadin }>

Vaadin Sampler

http://demo.vaadin.com/sampler/

Google

Vaadin Sampler

Home

All Samples

- UI Basics
  - Tooltips
  - Icons
  - Runo theme icons NEW
  - Error indicator
  - Progress indication NEW
  - JavaScript API NEW
  - Buttons
    - Push button
    - Link button
    - Checkbox
  - Links
    - Link
    - Link, configure window
    - Link, sized window
  - Texts
    - Label, plain text
    - Label, preformatted
    - Label, rich text
  - Embedding
    - Image NEW
    - Flash NEW
    - Web content NEW
- Value Input Components
  - Dates
    - Pop-up date selection

UI Basics

18 SAMPLES

Tooltips

Icons

Runo theme icons


Error indicator

Progress indication

JavaScript API

Push button

Link button


Checkbox

Link

Link, configure window

Link, sized window

Label, plain text

Label, preformatted

Label, rich text

Image

Flash

Web content

vaadin }>


Carrier
12:25 PM
100%
Mailboxes
4 of 6
Details
Vmail
Inbox (3)
Edit
sit@ipsum.com
Today
29225: Eirmod tempor invidunt ut sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur s...
ut@clita.com
Today
69605: Aliquyam erat, sed diam dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invid...
takimata@dolor.com
Today
29388: Dolor sit amet. Lorem ipsum ... Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod ...
nonumy@labore.com
Today
16940: Stet clita kasd gubergren, n... eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no s...
justo@eirmod.com
Today
75623: Magna aliquyam erat, sed sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur s...
elitr@aliquyam.com
Today
14630: Dolor sit amet, consetetur Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod ...
Updated 10/3/11 09:13
sea takimata sanctus est
a rebum. Stet clita kasd gubergren, no sea takimata sanctus
sum dolor sit amet, consetetur sadipscing elitr, sed diam
et dolore magna aliquyam erat, sed diam voluptua. At vero
a rebum. Stet clita kasd gubergren, no sea takimata sanctus
sum dolor sit amet, consetetur sadipscing elitr, sed diam
et dolore magna aliquyam erat, sed diam voluptua. At vero
a rebum. Stet clita kasd gubergren, no sea takimata sanctus
sum dolor sit amet, consetetur sadipscing elitr, sed diam
sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing
vidunt ut labore et dolore magna aliquyam erat, sed diam
Carrier
3:31 PM
Mailboxes
Accounts
Vmail
3
Hotvail
Vahoo!
300
vCloud
1
Updated 9/30/11 12:31
vaadin

User Inteface

Data Source

Theme

vaadin }>

Reindeer

Runo

Chameleon

Custom

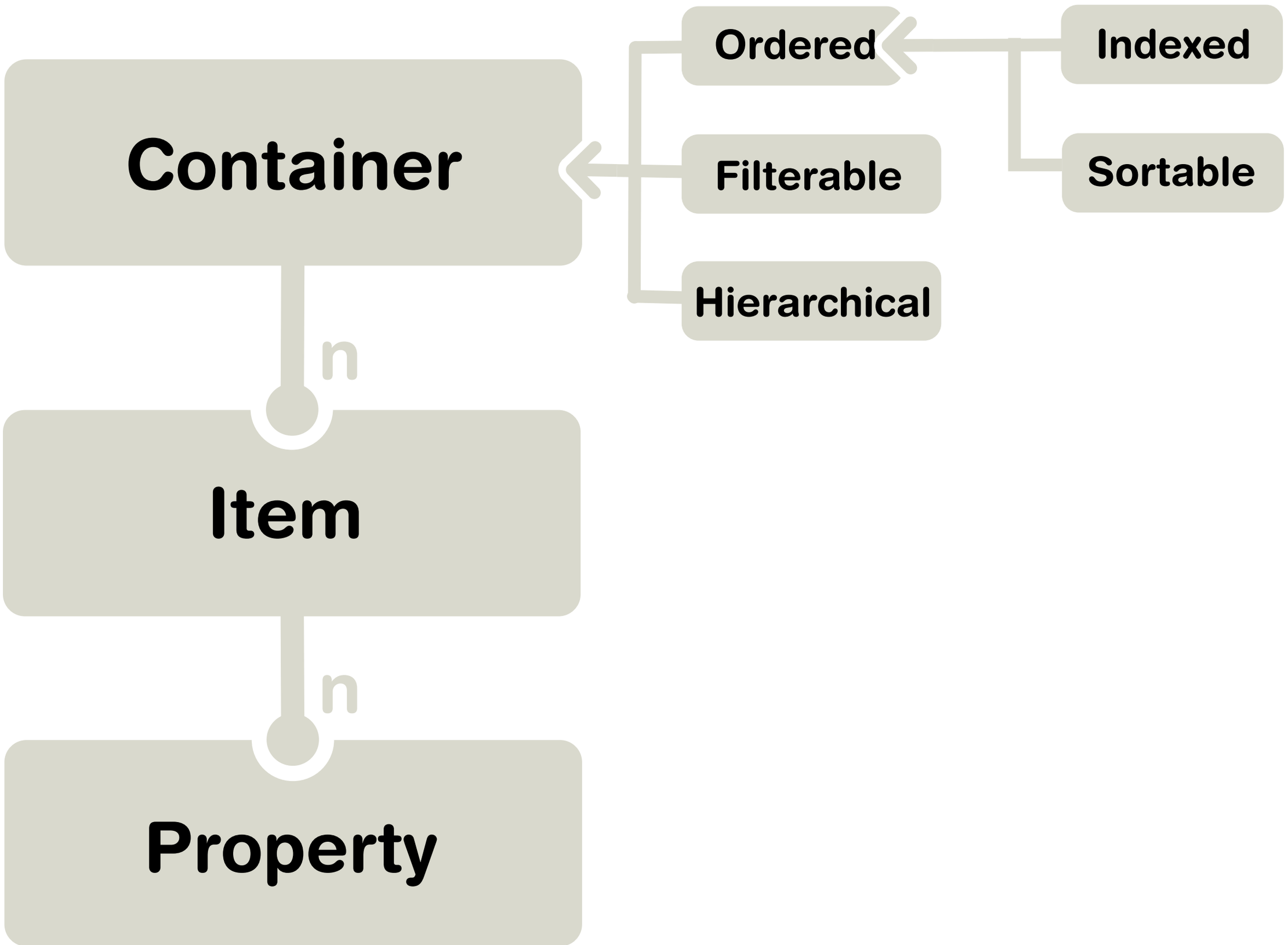
vaadin }>

User Inteface

Data Source

Theme

vaadin }>


Container
Ordered
Indexed
Filterable
Sortable
Hierarchical
n
Item
n
Property

vaadin

InMemory, Bean, Method, Collection, JDBC, JPA, Hibernate, TextFile, FileSystem, Properties, EclipseLink, Lucene, Mockups, GAE, ... Your's

vaadin }>

combine the best of

- Server-side RIA
- Google Web Toolkit

vaadin }>

combine the best of

- Server-side RIA
- Google Web Toolkit

vaadin }>

Layers of abstraction

	backend server	frontend server	RPC	browser	browser
	any language	**any language**	**json / xml**	**java**	**javascript**
Vaadin	required	required	*optional*	*optional*	*optional*
GWT	required	required	required	required	*optional*
ExtJS	required	required	required	X	required

vaadin }>

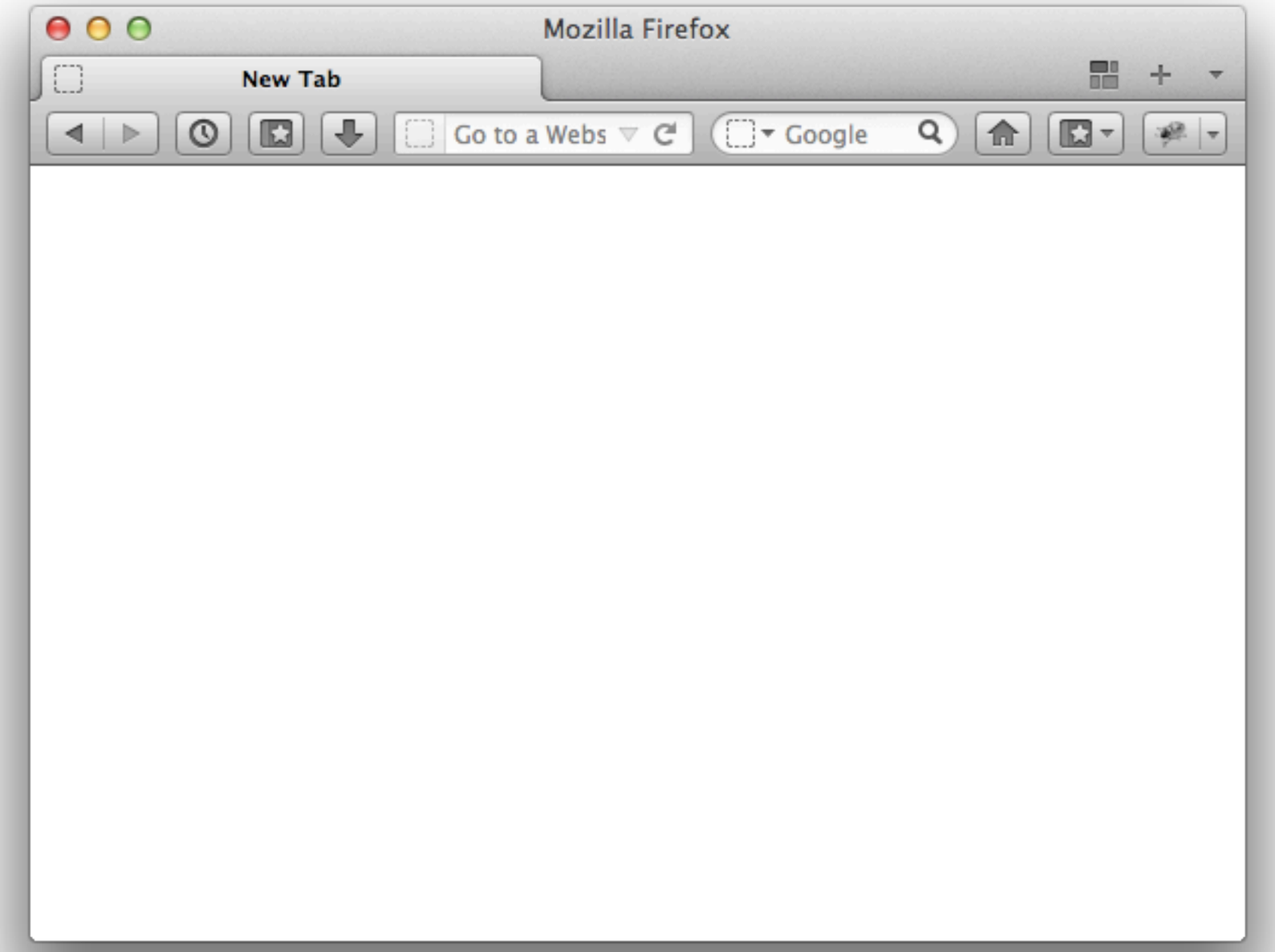
How does it work, really?

vaadin }>

```
name = new TextField("Name");
greetButton = new Button("Greet");

greetButton.addListener(new ClickListener() {
    public void buttonClick(ClickEvent event) {
        mainWindow.showNotification("Hi " + name);
    }
});
```

vaadin }>



- Initial HTML
- CSS (theme)
- Images
- JavaScript

830k total

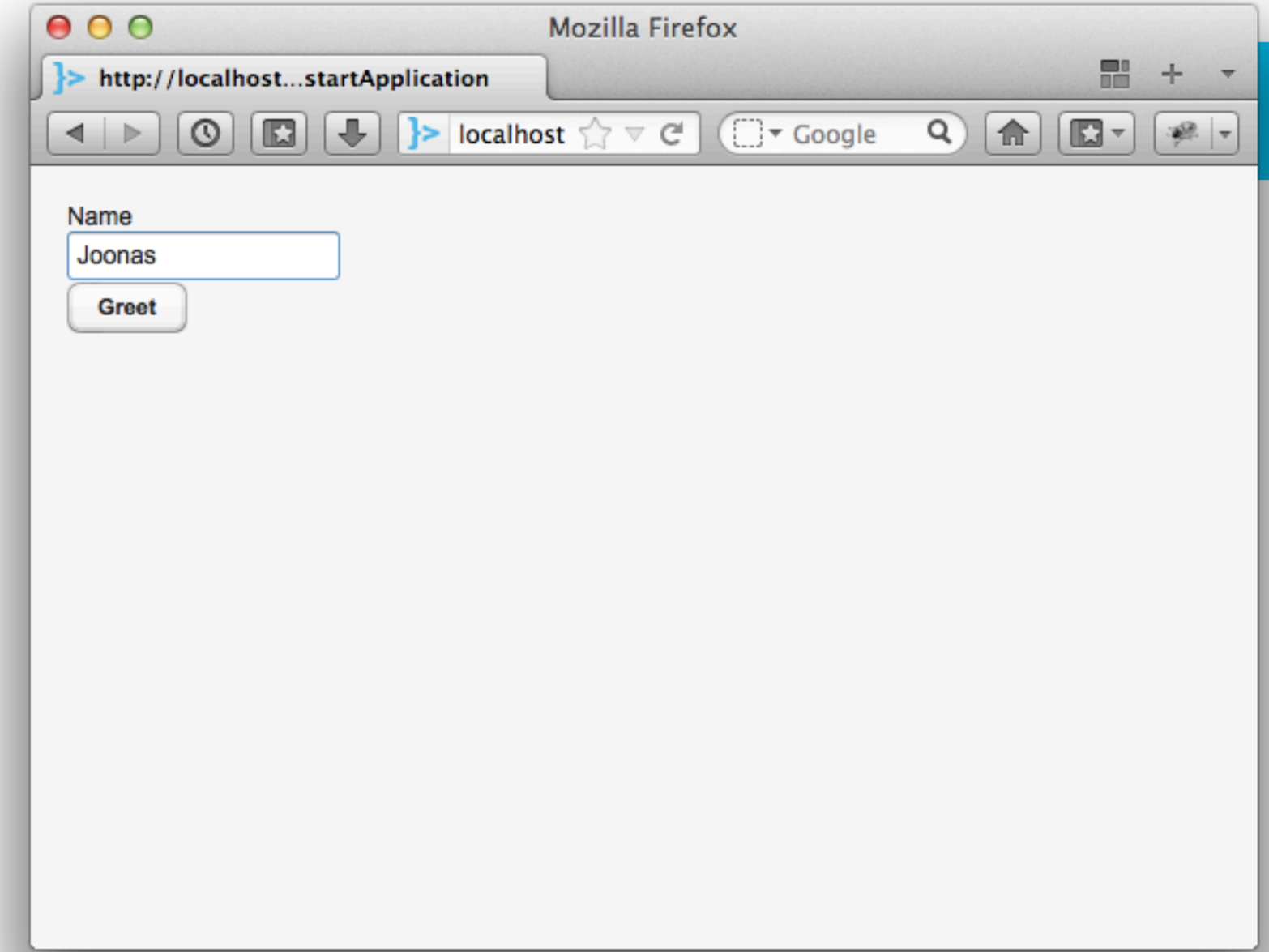
compress

250k

reduced widgetset

120k

vaadin



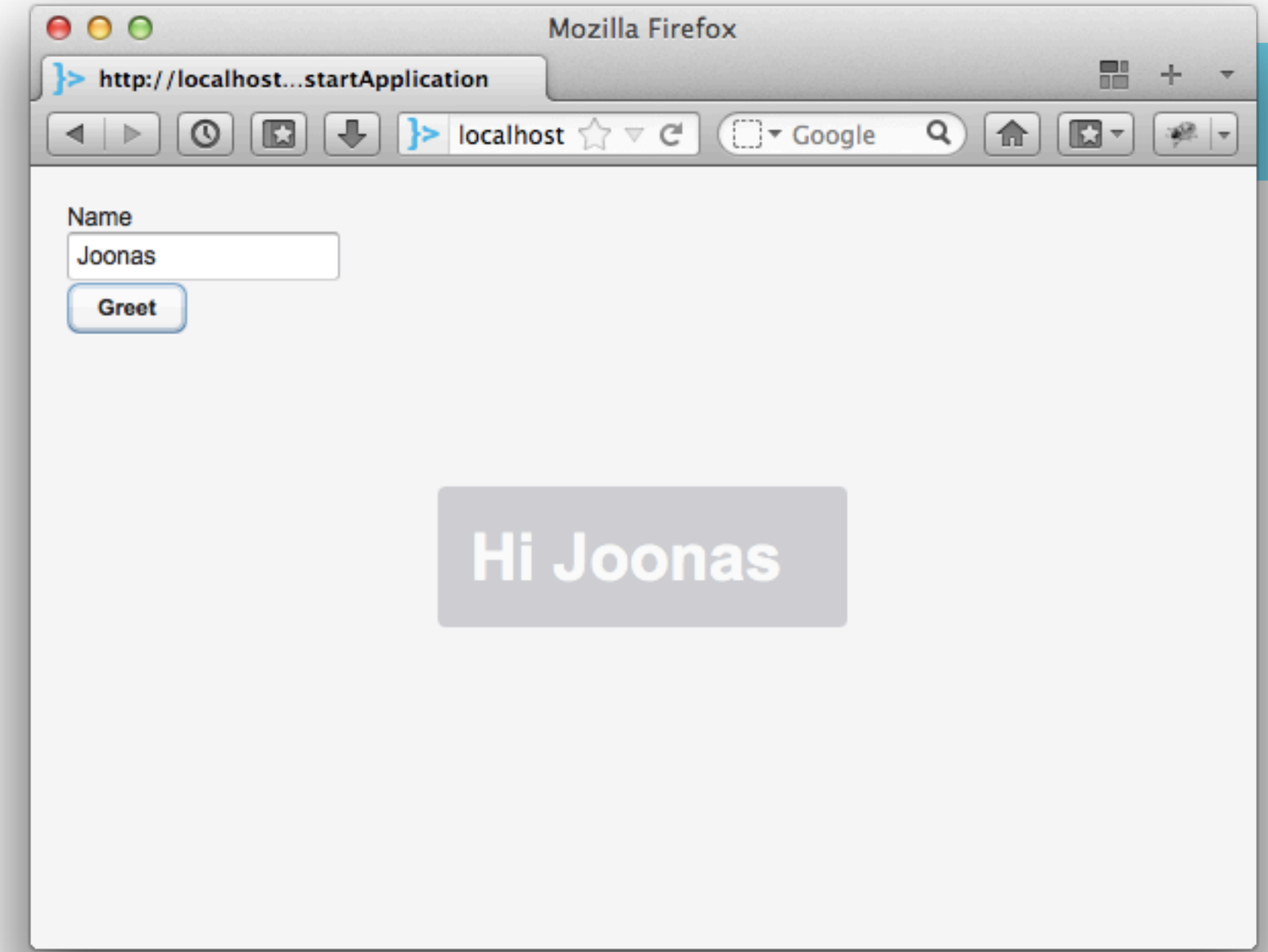
- name=”Joonas”
- button clicked

150 bytes

vaadin }>

```
public void buttonClick(ClickEvent event) {
    mainWindow.showNotification("Hi " + name);
}
```

vaadin }>



- name=”Joonas”
- button clicked

150 bytes

- Add notification

466 bytes

vaadin }>

Vaadin UI component architecture

“UI Component”
- Button, Table, Tree, ...
- Server-side data
- Full Java API

HTTP(S)

“Widget”
- Client-side peer for the component
- Runs on JavaScript

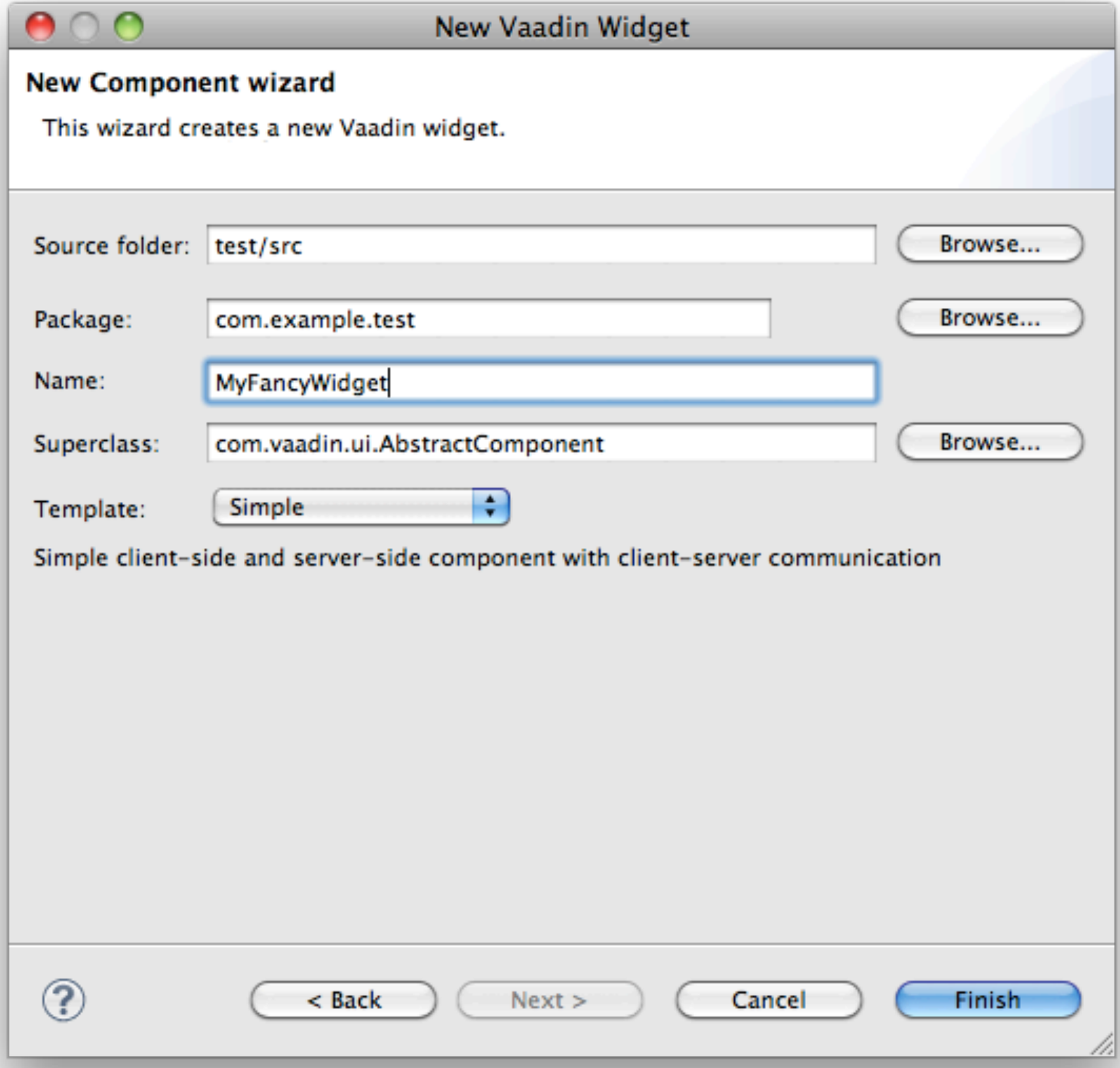
Java
- Compiled with JDK

Java
- Google Web Toolkit

vaadin }>

Creating new UI components is really easy

vaadin }>


New Vaadin Widget
New Component wizard
This wizard creates a new Vaadin widget.
Source folder:
test/src
Browse...
Package:
com.example.test
Browse...
Name:
MyFancyWidget
Superclass:
com.vaadin.ui.AbstractComponent
Browse...
Template:
Simple
Simple client-side and server-side component with client-server communication
< Back
Next >
Cancel
Finish

Implement two classes

Server-side

“UI Component”

- Define **API**
- Receive client events
- Send UI updates back

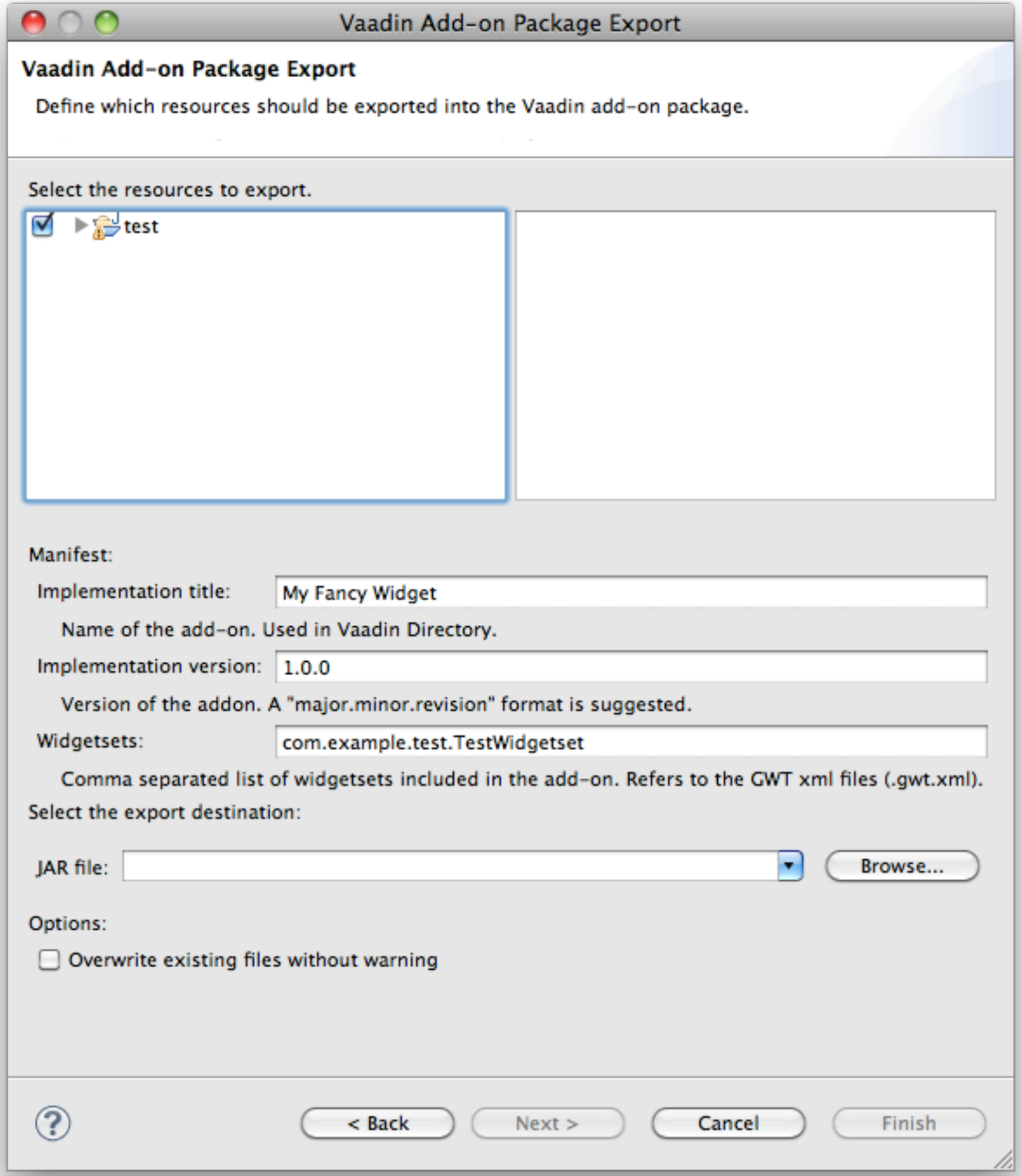
Automatic

Client-side

“Widget”

- **Render** to DOM
- Collect user events

vaadin }>


Vaadin Add-on Package Export
Vaadin Add-on Package Export
Define which resources should be exported into the Vaadin add-on package.
Select the resources to export.
test
Manifest:
Implementation title:
My Fancy Widget
Name of the add-on. Used in Vaadin Directory.
Implementation version:
1.0.0
Version of the addon. A "major.minor.revision" format is suggested.
Widgetsets:
com.example.test.TestWidgetset
Comma separated list of widgetsets included in the add-on. Refers to the GWT xml files (.gwt.xml).
Select the export destination:
JAR file:
Browse...
Options:
Overwrite existing files without warning
< Back
Next >
Cancel
Finish

Upload New Add-on

Select a category to post your new add-on to.

Note, that if you're updating a previous add-on, that is done by editing the add-on from the list above.

UI Components ☑

Server-side and/or client-side UI components

Themes ☐

Themes for Vaadin applications

Data Components ☐

Components related to the Vaadin data model, e.g. Container or Validator implementations

Tools ☐

Tools for Vaadin developers

Miscellaneous ☐

Other Vaadin add-ons

Upload Add-on Package

vaadin

Directory

Search Add-ons

Browse

- All
- UI Components
- Data Components
- Themes
- Tools
- Miscellaneous

Guest
Authoring

Subscribe RSS
Help
FAQ
Feedback

Most Recent | Highest Rated | Top Downloads

Showing CERTIFIED STABLE BETA EXPERIMENTAL

« Previous Next » 1 2 3 4 5 6 7 8 9 10 11 12 13 14 204 Results

EasyUploads
In UI Components by Matti Tahvonen

Use file uploads as fields in Form, upload multiple files at once - easily!

Version 0.4.2 BETA ★★★★½ 2 304

OpenLayers Wrapper
In UI Components by Matti Tahvonen

Vaadin server side components that wrap essential OpenLayers objects

Version 0.4.0 EXPERIMENTAL ★★★★☆ 2 142

I18N4Vaadin
In Miscellaneous by Petter Holmström

A small add-on for creating localized applications

Version 0.9.0 BETA ★★★★★ 1 10

Navigator
In UI Components by Joonas Lehtinen

Navigator is an easy to use view manager that supports lazy initialization, bookmarking and multiple browser windows.

Version 0.3 EXPERIMENTAL ★★★★½ 3 234

ConfirmDialog
In UI Components by Run Uilder

A versatile confirm dialog for Vaadin

Version 1.1.0 BETA ★★★½☆ 6 676

CustomField
In UI Components by Henri Sara

A form field whose presentation and logic can be customized

Version 0.8.2 BETA ★★★★★ 8 1075

Drawer
In UI Components by Henrik Paul

An animated component to hide or show information

Transactional Container
In Data Components by Tommi Laukkanen

Transactional Container offers same base features as

Directory

Search Add-ons

Browse

All
UI Components
Data Components
Themes
Tools
Miscellaneous

Guest
Authoring

Subscribe RSS
Help
FAQ
Feedback

PaperStack

In UI Components by Tomi Virkki ★★★★★ 11 194

Report this add-on

Version 0.8.1 (latest)

Maturity EXPERIMENTAL
License Apache License 2.0
Vaadin 6.2 upwards

Browser Compatibility
3
5 6
7 8
10
3 4 5

Overview

PaperStack is a component container whose subcomponents are presented sequentially, one subcomponent at a time. User can switch between the subcomponents by mouse dragging the upper right corner of a view revealing the underlying subcomponent simultaneously. The transition effect simulates leafing through a stack of papers.

Highlights

Code Example


Screenshot 2


Screenshot 1

Release notes

Download Now
Version 0.8.1 (86 kB)

Maven POM

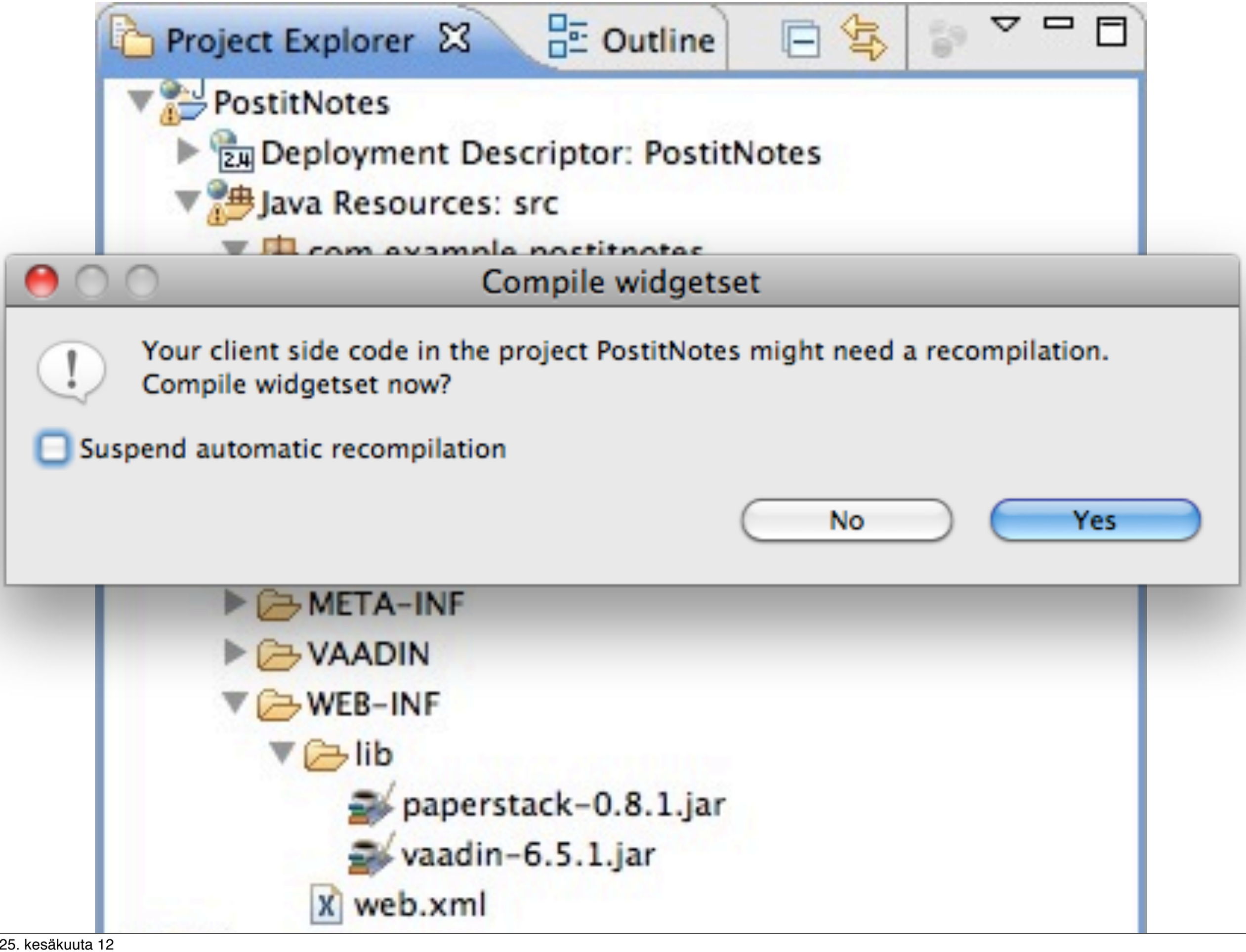
Related Links

Discussion Forum
Online Demo
Source Code

Share

More...

Permalink to this add-on:
http://vaadin.com/addon/paperstack


Project Explorer
Outline
PostitNotes
Deployment Descriptor: PostitNotes
Java Resources: src
Compile widgetset
Your client side code in the project PostitNotes might need a recompilation.
Compile widgetset now?
Suspend automatic recompilation
No
Yes
META-INF
VAADIN
WEB-INF
lib
paperstack-0.8.1.jar
vaadin-6.5.1.jar
web.xml

```
PaperStack notes = new PaperStack();

   @Override
   public void init() {

       // === Layout ============================================
       HorizontalLayout lo= new HorizontalLayout();
       Window mainWindow = new Window("Postitnotes Application", lo);
       lo.setSizeFull();
       lo.addComponent(notes);
       lo.setComponentAlignment(notes, Alignment.MIDDLE_CENTER);
       setMainWindow(mainWindow);
       notes.setWidth("350px");
       notes.setHeight("350px");

       // === Note 1 ============================================
       notes.addComponent(new Label("<h1>TODO / Today</h1><div style='font-size: 24px'>" +
          "<p>Enjoy the conference...</p></div>", Label.CONTENT_XHTML), "#fef49c");

       // === Note 2 ============================================
       notes.addComponent(new Label("<h1>TODO / Tomorrow</h1><div style='font-size: 24px'>" +
          "<p>Learn Vaadin!</p></div>", Label.CONTENT_XHTML), "#b2ffa1");

       // === Note 3 ============================================
       notes.addComponent(new Label("<div style='font-size: 60px'><center><br/><br/><br/>" +
          "DOUBLE<br/><br/><br/>SPEED</center></div>", Label.CONTENT_XHTML), "#b2ffa1");

       // === Note 4 ============================================
       com.vaadin.ui.RichTextArea rta = new RichTextArea();
       rta.setSizeFull();
       notes.addComponent(rta);
       rta.setValue("<span style='font-size: 35pt; color: green;'>You can use any " +
          "Vaadin components here...</span>");
   }
```

vaadin }>

embrace java

vaadin }>

Any JVM Language

vaadin }>

Scala with Scaladin add-on

```
val layout =
    new VerticalLayout(width = 100 pct, height = 100 pct) {
        add(new Label(content = "Persons"))
        add(new Table(width = 100 pct, height = 100 pct),
            ratio = 1)
        add(new HorizontalLayout(spacing = true) {
            add(new Button("Edit selected", _ => editClicked()))
            add(new Button("Add new", _ => addNewClicked()))
        })
    }
getMainWindow.setContent(layout)
```

vaadin }>

6..

Internet Explorer
Chrome
Firefox
Safari
Opera
iOS
Android

vaadin }>

No browser plugins

Nothing to install

vaadin }>

Servlet Portlet (most) clouds

vaadin }>

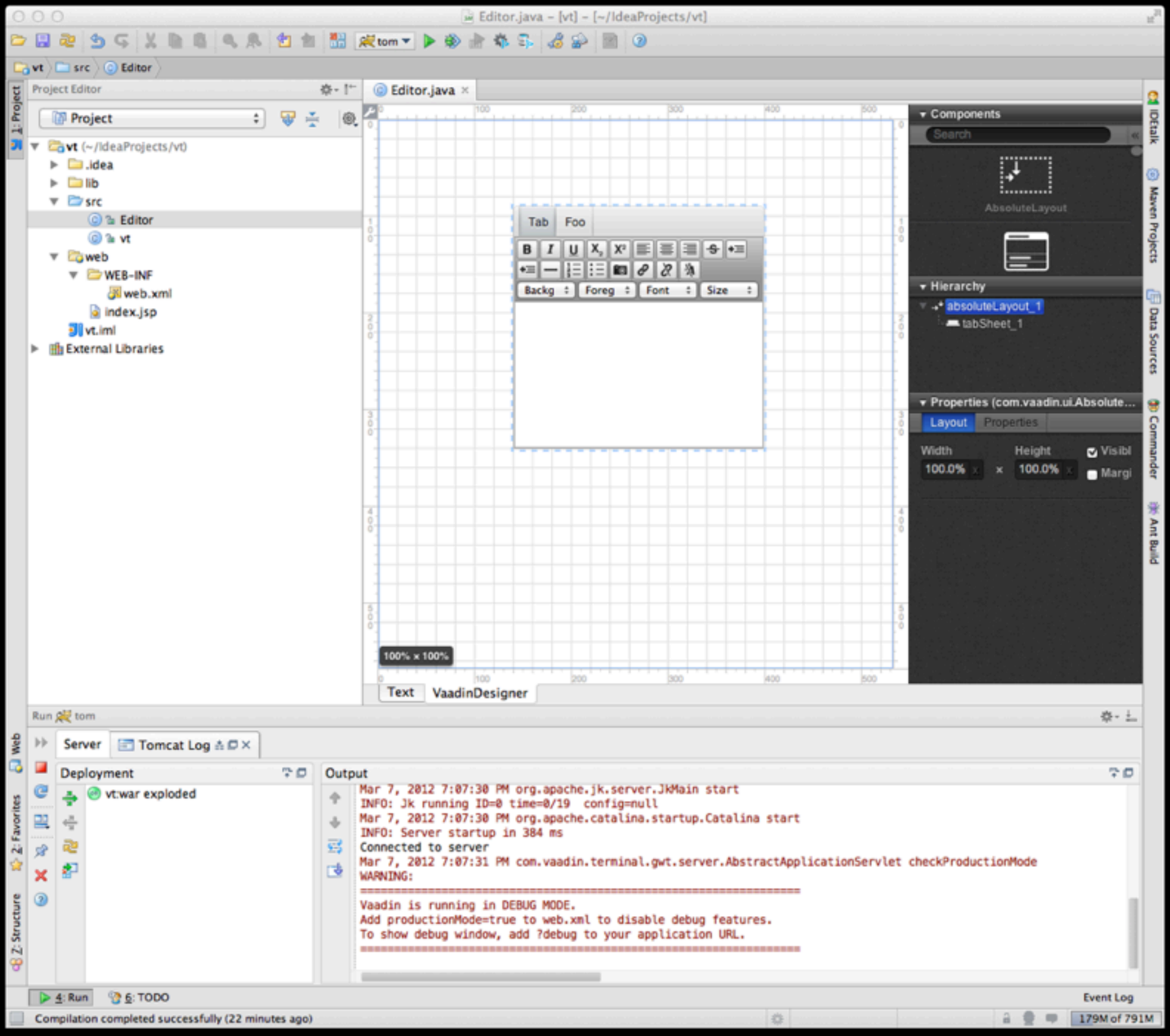
Apache Tomcat, version 4.1 or later
Oracle WebLogic Server, version 9.2 or later
Oracle WebLogic Portal, version 10gR3
IBM WebSphere Application Server, version 6.1 or later
IBM WebSphere Portal, version 6.1 and 7.0
JBoss Application Server, version 3.2.8 or later
Jetty, version 5 or later
Glassfish, version 2 or later
Liferay Portal 5.2 or later
GateIn Portal 3.1
eXo Platform 3
Google App Engine

Vaadin supports Java Servlet API 2.3
and JSR-168 and JSR-286 Portlet
Specifications and should work with
any Java application server that conforms
to these standards.

vaadin }>

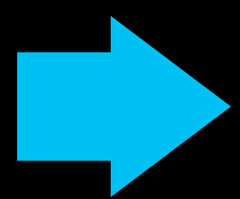
Eclipse
IntelliJ IDEA
Netbeans
Maven
Ant
Spring Roo
...

vaadin }>


Editor.java – [vt] – [~/IdeaProjects/vt]
vt
src
Editor
Project Editor
Project
vt (~/IdeaProjects/vt)
.idea
lib
src
Editor
vt
web
WEB-INF
web.xml
index.jsp
vt.iml
External Libraries
Editor.java
Tab
Foo
Backg
Foreg
Font
Size
100% x 100%
Text
VaadinDesigner
Components
Search
AbsoluteLayout
Hierarchy
absoluteLayout_1
tabSheet_1
Properties (com.vaadin.ui.Absolute...
Layout
Properties
Width
Height
100.0%
Visibl
Margi
IDEtalk
Maven Projects
Data Sources
Commander
Ant Build
Run tom
Server
Tomcat Log
Deployment
vt:war exploded
Output
Mar 7, 2012 7:07:30 PM org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/19 config=null
Mar 7, 2012 7:07:30 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 384 ms
Connected to server
Mar 7, 2012 7:07:31 PM com.vaadin.terminal.gwt.server.AbstractApplicationServlet checkProductionMode
WARNING:
Vaadin is running in DEBUG MODE.
Add productionMode=true to web.xml to disable debug features.
To show debug window, add ?debug to your application URL.
1: Project
2: Favorites
Web
7: Structure
4: Run
6: TODO
Event Log
Compilation completed successfully (22 minutes ago)
179M of 791M

Maven

```
mvn archetype:generate
-DarchetypeGroupId=com.vaadin
-DarchetypeArtifactId=
vaadin-archetype-clean
```

mvn package  **yourproject-1.0.war**

vaadin }>

Vaadin Framework 7

vaadin }>

History

When	What
2001	Millstone Framework is born
2002	Released as Open Source
2005	Ajax based rendering
2008	Google Web Toolkit integrated
2009	Renamed to Vaadin
2010	Add-ons

Server-side API compatible

Heavily under construction

[to be released on Q4]

vaadin }>

Current Development Milestones

Milestone: Vaadin 6.8

5 weeks late *(04/15/12 18:00:00)*

Number of tickets: closed: 41 active: 9 Total: 50

Milestone: Vaadin 6.7.9

Due in 4 days (05/21/12 18:00:00)

Number of tickets: closed: 10 active: 5 Total: 15

Milestone: Vaadin 7.0.0.alpha3

Due in 3 weeks (06/05/12 16:00:00)

Number of tickets: closed: 0 active: 65 Total: 65

- Navigation API for applications
- Make theming more flexible
- Improve session management and related messages
- Javascript callbacks to server
- Wrapper for Javascript components
- Plug-in support for Roots (aka components without a UI)
- Plug-in support for ApplicationServlet
- Ease embedding of Vaadin applications into JSP and JSF applications

vaadin.com/roadmap

vaadin }>

Empower Developers

Embrace Extendability

Clean Up

Empower Developers

View Management Redesign

- Remove "browser window"
- HttpRequest access
- View management API
- Simplified Application API by removing unnecessary abstractions
- Multitab support by default

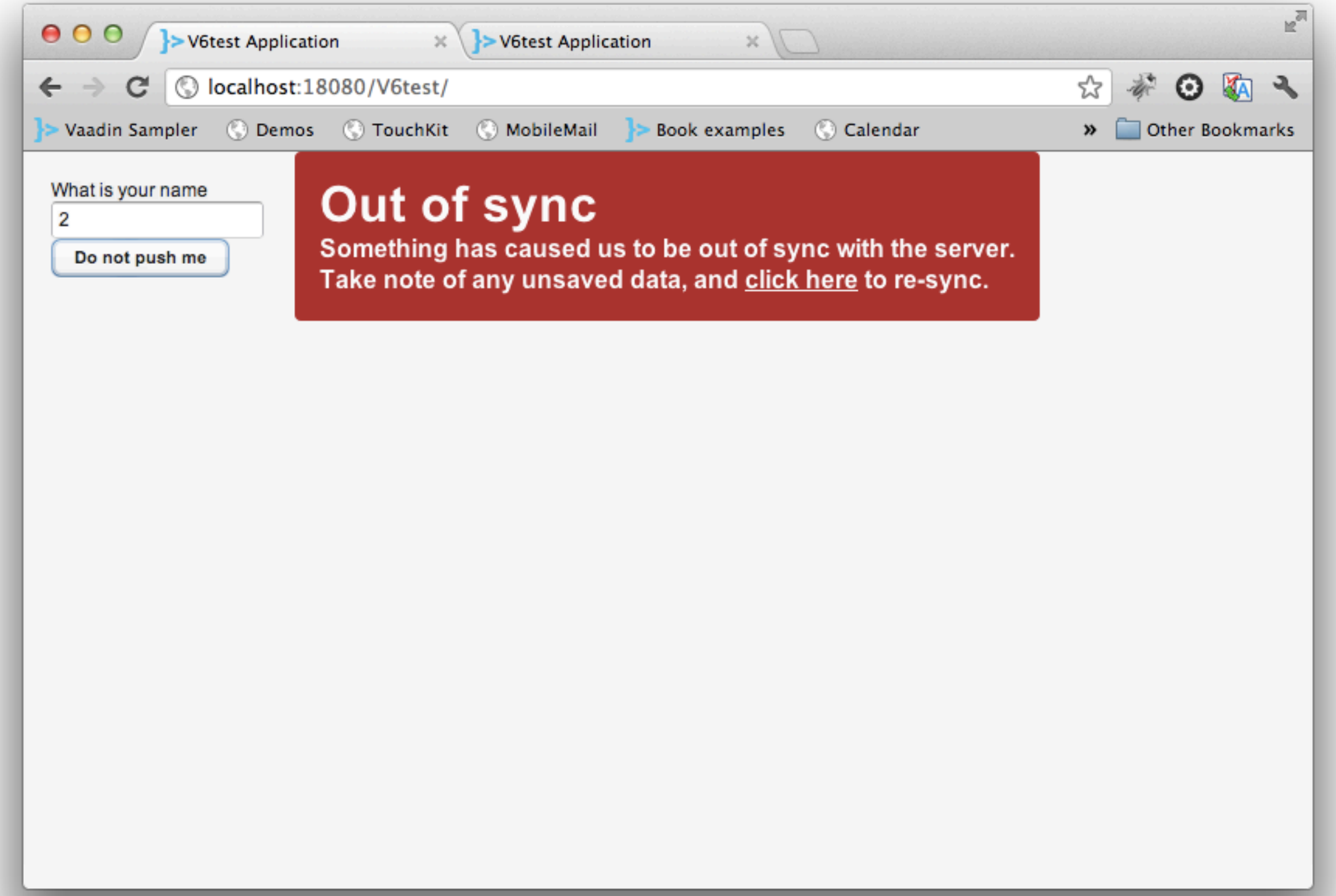
```
public class Vaadin6testApplication extends Application {

   public void init() {
      Window mainWindow = new Window("V6test Application");
      setMainWindow(mainWindow);

      mainWindow.addComponent(new TextField("What is your name"));
      mainWindow.addComponent(new Button("Do not push me"));
   }

}
```

vaadin }>


V6test Application
V6test Application
localhost:18080/V6test/
Vaadin Sampler
Demos
TouchKit
MobileMail
Book examples
Calendar
Other Bookmarks
What is your name
2
Do not push me
Out of sync
Something has caused us to be out of sync with the server.
Take note of any unsaved data, and click here to re-sync.

```
public class V6testApplication extends Application {

    public void init() {
        setMainWindow(createWindow());
    }

    public Window getWindow(String name) {
        Window window = super.getWindow(name);
        if (window == null) {
            window = createWindow();
            window.setName(name);
            addWindow(window);
        }
        return window;
    }

    private Window createWindow() {
        Window window = new Window("V6test Application");
        window.addComponent(new TextField("What is your name"));
        window.addComponent(new Button("Do not push me"));
        return window;
    }

}
```

vaadin }>

```
public class Vaadin7testRoot extends Root {

    public void init(WrappedRequest request) {
        addComponent(new TextField("What is your name"));
        addComponent(new Button("Do not push me"));
    }

}
```

vaadin }>

Warning: Naming might be changed from “Root” to “Region”

vaadin }>

```
public class Vaadin6Application extends Application {

    public void init() {
        Window window = new Window("V6test Application");
        setMainWindow(window);

        final Label label = new Label();
        final UriFragmentUtility fragment = new UriFragmentUtility();
        window.addComponent(fragment);

        window.addComponent(label);

        fragment.addListener(new UriFragmentUtility.FragmentChangedListener() {
            public void fragmentChanged(FragmentChangedEvent source) {
                label.setValue(fragment.getFragment());
            }
        });
    }

}
```

vaadin }>

```
public class V7testRoot extends Root {

    public void init(WrappedRequest request) {
        addComponent(new Label(getFragment()));
    }

}
```

vaadin }>

```
public class V7testRoot extends Root {

   public void init(WrappedRequest request) {

      final Label label = new Label(getFragment());
      addComponent(label);

      addListener(new FragmentChangedListener() {
         public void fragmentChanged(FragmentChangedEvent event) {
            label.setValue(getFragment());
         }
      });
   }

}
```

vaadin }>

Most of the changes already available Alpha 1

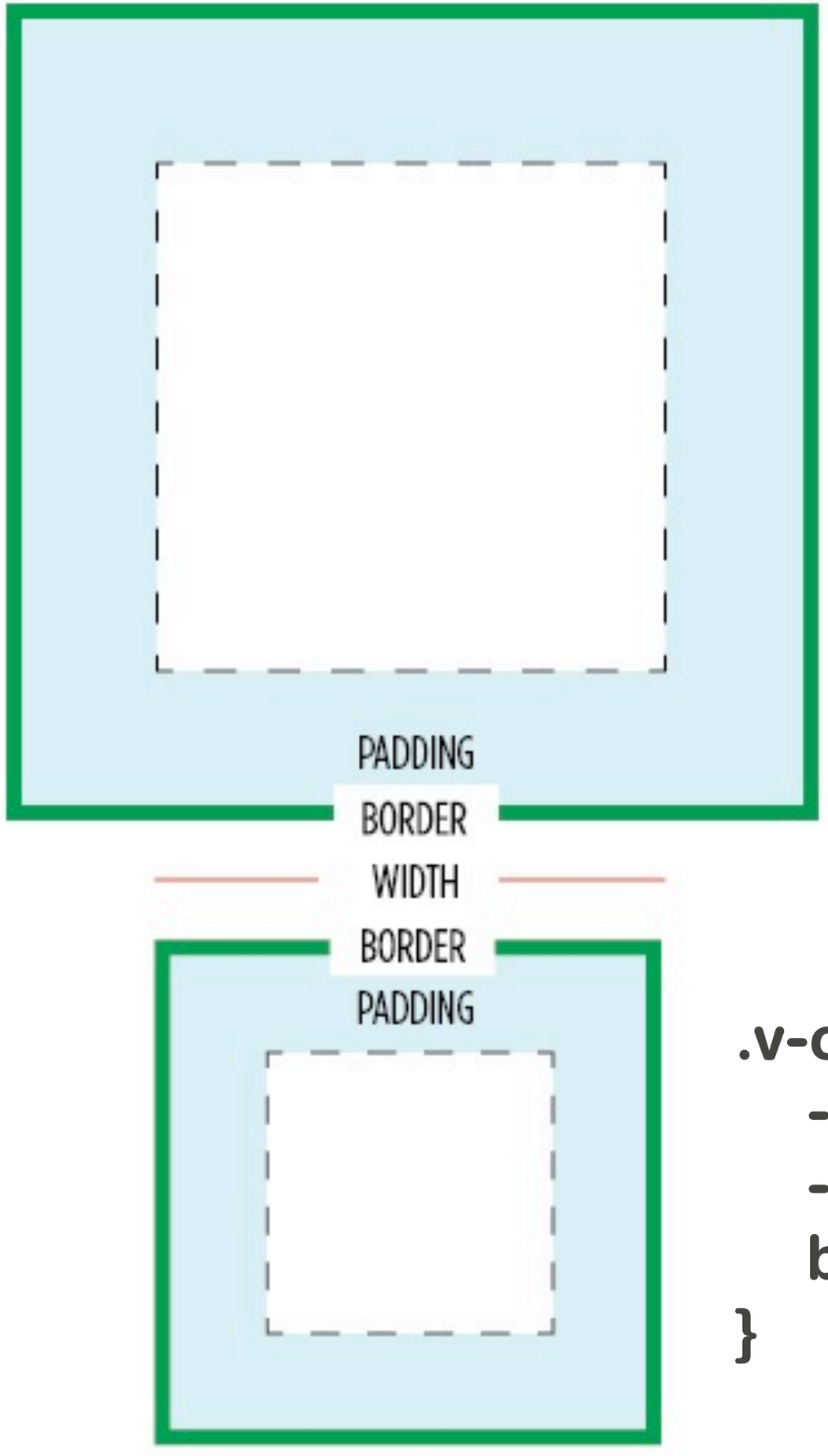
View management API in Alpha 3 to be released in June

vaadin }>

Empower Developers

Full CSS Control

- Layout calculations are managed by browser
  - Improved performance
  - Support border/margin/...
  - Drop support for IE6/IE7
- CSS abstraction language
- Inject CSS from Java
- Per component selectors




```
.v-connector {
    -webkit-box-sizing: border-box;
    -moz-box-sizing: border-box;
    box-sizing: border-box;
}
```

vaadin }>

```
private final ElementResizeListener listener = new ElementResizeListener() {
    public void onElementResize(ElementResizeEvent e) {
        int buttonWidth = getLayoutManager().getOuterWidth(e.getElement());
        buttonWidth -= getLayoutManager().getMarginRight(e.getElement());
        getWidget().adjustButtonSpace(buttonWidth);
    }
};

@Override
protected void init() {
    Element button = getWidget().getWidget(1).getElement();
    getLayoutManager().addElementResizeListener(button, listener);
}

@Override
public void onUnregister() {
    Element button = getWidget().getWidget(1).getElement();
    getLayoutManager().removeElementResizeListener(button, listener);
}
```

vaadin }>

Alpha 2 implements lots of layout performance improvements, but work is still unfinished.

vaadin }>

vaadin }>

Variables & functions

.scss .sass

```scss
$blue: #3bbfce;
$margin: 16px;

.content-navigation {
  border-color: $blue;
  color:
    darken($blue, 9%);
}

.border {
  padding: $margin / 2;
  margin: $margin / 2;
  border-color: $blue;
}
```

```css
/* CSS */

.content-navigation {
  border-color: #3bbfce;
  color: #2b9eab;
}

.border {
  padding: 8px;
  margin: 8px;
  border-color: #3bbfce;
}
```

Mixins

.scss .sass

```
@mixin table-base {
  th {
    text-align: center;
    font-weight: bold;
  }
  td, th {padding: 2px}
}

@mixin left($dist) {
  float: left;
  margin-left: $dist;
}

#data {
  @include left(10px);
  @include table-base;
}
```

```
/* CSS */

#data {
  float: left;
  margin-left: 10px;
}
#data th {
  text-align: center;
  font-weight: bold;
}
#data td, #data th {
  padding: 2px;
}
```

Nesting

.scss .sass

```
table.hl {
  margin: 2em 0;
  td.ln {
    text-align: right;
  }
}

li {
  font: {
    family: serif;
    weight: bold;
    size: 1.2em;
  }
}
```

```
/* CSS */

table.hl {
  margin: 2em 0;
}
table.hl td.ln {
  text-align: right;
}

li {
  font-family: serif;
  font-weight: bold;
  font-size: 1.2em;
}
```

Selector Inheritance

.scss .sass

```
.error {
  border: 1px #f00;
  background: #fdd;
}
.error.intrusion {
  font-size: 1.3em;
  font-weight: bold;
}

.badError {
  @extend .error;
  border-width: 3px;
}
```

```
/* CSS */

.error, .badError {
  border: 1px #f00;
  background: #fdd;
}

.error.intrusion,
.badError.intrusion {
  font-size: 1.3em;
  font-weight: bold;
}

.badError {
  border-width: 3px;
}
```

Empower Developers

Glue for Integrations

- Publish JavaScript API from the server-side
- Integrate easily with the rest of the web page
- Tag lib to ease integration with JSP/JSF
- Centralized shared resource loading (JS/CSS)

Empower Developers

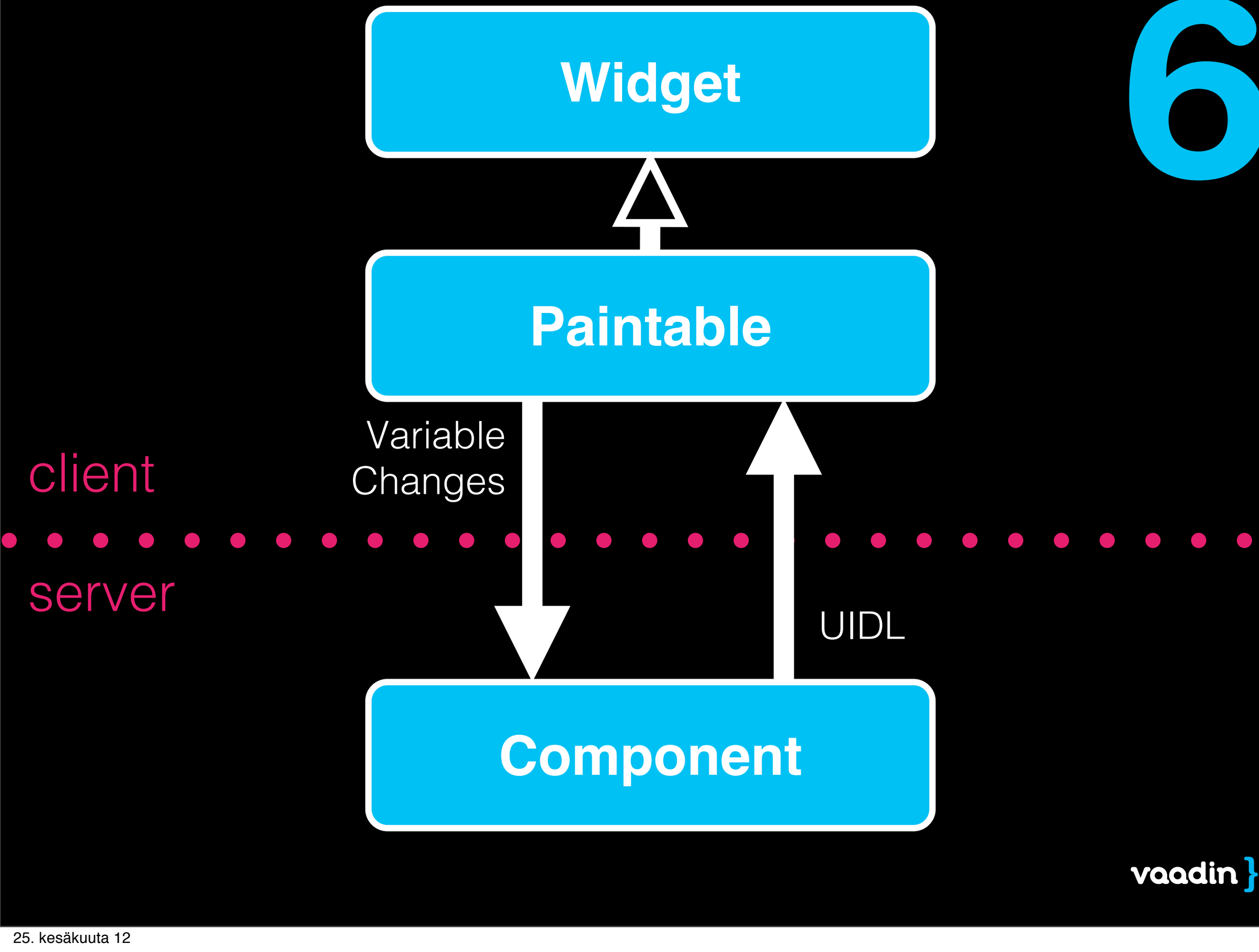
Live Translations

- Built in translator API
- Custom translation stores
- Translated transparently on client-server communication
- Switch languages on fly
- Strings and resources

Embrace Extendability

Easier Client-Server Communications

- Client-server RPC API
- Shared state API
- Phase out “variables” and “attributes”


Widget
Paintable
Variable
Changes
client
server
UIDL
Component

7

Widget

Connector

client

server

RPC

State

Component

vaadin }>

State

- Bean with most Java types: integers, Strings, lists, maps, ...
- All server-side changes automatically sent to client.
- Connectors can access all states of children and parent components

vaadin }>

RPC

client to server example

```
public interface ButtonRpc extends ServerRpc {
      public void click(MouseEventDetails details);
}
```

```
private ButtonRpc rpc =
RpcProxy.create(ButtonRpc.class, this);

public void onClick(ClickEvent event) {
    rpc.click(
        new MouseEventDetails(event));
}
```

```
private ButtonRpc rpc = new ButtonRpc() {
    public void click(
        MouseEventDetails details) {
            // do stuff
    }
};

public Button() {
    registerRpc(rpc);
}
```

client server

vaadin }>

Already done in Alpha 2, start trying these out today...

vaadin }>

Embrace Extendability

New Extension Points

- ApplicationServlet plug-ins
- “Non-widget” add-ons
- Component decorator for extending without extend
- Server-push plug-in API
  - Plug-in for push
- State storage plug-ins

Embrace Extendability

Generic JavaScript Widget Wrapper

- Convert existing 3rd party widgets to Vaadin
- GWT compilation not required

Embrace Extendability

Ease Common Extensions

- Allow easy packaging of CSS and images with server-side compositions
- Simplify Component Container API to make implementation easier

Clean Up

Data Binding and Form redesign

- Form binding and layout are cleanly separated
- Explicit property convertters
- Two phase commit support
- Built-in JSR-303 validations
- Backwards compatibility helper as add-on

Converters

vaadin }>

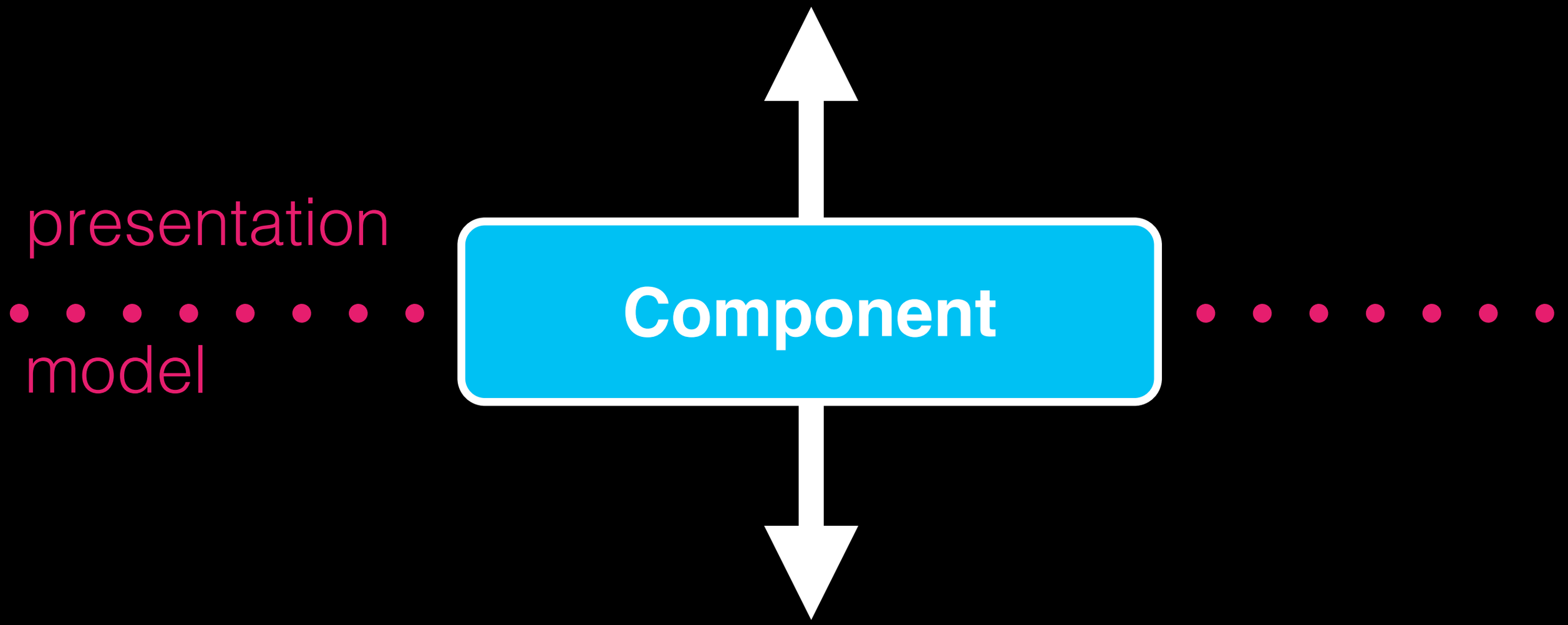
```java
public class Name {
    private String firstName;
    private String lastName;

    public Name(String firstName, String lastName) {
        this.firstName = firstName;
        this.lastName = lastName;
    }

    // Setters and getters

}
```

“Joonas Lehtinen”



firstName = “Joonas”
lastName = “Lehtinen”

vaadin }>

```java
public class StringToNameConverter implements Converter<String, Name> {

    public Name convertToModel(String text, Locale locale)
            throws ConversionException {
        if (text == null) {
            return null;
        }
        String[] parts = text.split(" ");
        if (parts.length != 2) {
            throw new ConversionException("Can not convert text to a name: "
                    + text);
        }
        return new Name(parts[0], parts[1]);
    }

    public String convertToPresentation(Name name, Locale locale)
            throws ConversionException {
        if (name == null) {
            return null;
        } else {
            return name.getFirstName() + " " + name.getLastName();
        }
    }

    public Class<Name> getModelType() {
        return Name.class;
    }

    public Class<String> getPresentationType() {
        return String.class;
    }
}
```

vaadin }>

```
Name name = new Name("Rudolph", "Reindeer");

final TextField textField = new TextField("Name");

textField.setConverter(new StringToNameConverter());

// ....

Name name = (Name) textField.getConvertedValue();
```

```
public class MyStringToDoubleConverter extends
StringToDoubleConverter {

    @Override
    protected NumberFormat getFormat(Locale locale) {
        NumberFormat format = super.getFormat(locale);
        format.setGroupingUsed(false);
        format.setMaximumFractionDigits(3);
        format.setMinimumFractionDigits(3);
        return format;
    }
}
```

```
public class MyConverterFactory extends DefaultConverterFactory {

    @Override
    protected <PRESENTATION, MODEL>
        Converter<PRESENTATION, MODEL> findConverter(
            Class<PRESENTATION> presentationType,
            Class<MODEL> modelType) {

        // Handle String <-> Double
        if (presentationType == String.class &&
            modelType == Double.class) {
            return (Converter<PRESENTATION, MODEL>)
                new MyStringToDoubleConverter();
        }

        // Let default factory handle the rest
        return super.findConverter(presentationType, modelType);
    }
}

getApplication().setConverterFactory(new MyConverterFactory());
```

vaadin

```
table.setConverter(PERCENT_PROPERTY,
  new StringToNumberConverter() {
    @Override
    protected NumberFormat getFormat(Locale locale) {
        return NumberFormat.getPercentInstance(locale);
    }
});
```

```
table.setConverter(CURRENCY_PROPERTY,
  new StringToNumberConverter() {
    @Override
    protected NumberFormat getFormat(Locale locale) {
        return NumberFormat.getCurrencyInstance(locale);
    }
});
```

vaadin }>

Validation

vaadin }>

```java
public class Person {

    @Size(min = 5, max = 50)
    private String name;

    @Min(0)
    @Max(100)
    private int age;

    // + constructor + setters + getters
}
```

vaadin

```
Person person = new Person("John", 26);
BeanItem<Person> item = new
    BeanItem<Person>(person);

TextField firstName = new TextField("First name",
        item.getItemProperty("name"));

firstName.addValidator(new
    BeanValidator(Person.class, "name"));
```

vaadin

FieldGroup

vaadin }>

```
public class Employee {
   String firstName;
   String lastName;
   double salary;
   Date birthDate;

      // Getters, setters, …
}

Form form = new Form();

form.setItemDataSource(
    new BeanItem<Employee>(employee));
```

vaadin }>

Birth Date 11/20/75

First Name David

Last Name Smith

Salary 100000.0

```
Form form = new Form() {
    @Override
    protected void attachField(Object propertyId, Field field) {
        // Some custom logic for placing components
    }
};
```

vaadin }>

```
form.setFormFieldFactory(new FormFieldFactory() {

        public Field createField(Item item, Object propertyId,
              Component uiContext) {

           if ("salary".equals(propertyId)) {
              TextField tf = new TextField();
              tf.setWidth("7em");
              return tf;
           } else if ("birthDate".equals(propertyId)) {
              DateField df = new DateField();
              df.setResolution(DateField.RESOLUTION_DAY);
              return df;
           } else {
              // ..
           }

           return DefaultFieldFactory.createFieldByPropertyType(item
                 .getItemProperty(propertyId).getType());
        }
     });
```

vaadin }>

```
FormLayout form = new FormLayout() {

    TextField firstName = new TextField("First name");
    TextField lastName = new TextField("Last name");
    TextField salary = new TextField("Salary");
    DateField birthDate = new DateField("Birth date");

    {
        birthDate.setResolution(Resolution.DAY);
        addComponent(firstName);
        addComponent(lastName);
        addComponent(birthDate);
        addComponent(salary);
    }
};
```

vaadin }>

```
final BeanFieldGroup<Employee> fieldGroup =
      new BeanFieldGroup<Employee>(Employee.class);

fieldGroup.bindMemberFields(form);

fieldGroup.setItemDataSource(new BeanItem<Employee>(employee));
```

vaadin }>

First name David

Last name Smith

Birth date 11/20/75

Salary 100,000

```
GridLayout form = new GridLayout(2,2) {

  TextField firstName = new TextField("First name");
  TextField lastName = new TextField("Last name");
  TextField salary = new TextField("Salary");
  DateField birthDate = new DateField("Birth date");

  {
    birthDate.setResolution(Resolution.DAY);
    setSpacing(true);
    addComponent(firstName);
    addComponent(lastName);
    addComponent(birthDate);
    addComponent(salary);
  }
};
```

vaadin }>

First name

David

Last name

Smith

Birth date

11/20/75

Salary

100,000

Client-side

Clean Up

- Review widgetset API
- Ease client-side logic
- Decouple communication from core widgets thoughtout Vaadin 7 series
  - Client-side compositions
  - Support GWT Designer
- Better support for multiple Vaadin apps per Page

Improved session management

Clean Up

- Kill all “out-of-sync” errors
- Default to one server ~~Window~~ Root per page
- Default to keeping sessions live while page is shown

HTML

Clean Up

- Simpler DOM as IE6/IE7 is not supported any more
- WAI-ARIA support
- Built in HTML sanitation to improve security

Old API

Clean Up

- Replace integer constants with enums
- Remove the unnecessary Container.Ordered interface
- Remove deprecated APIs

Empower Developers

Embrace Extendability

Clean Up

Changes to roadmap are expected.

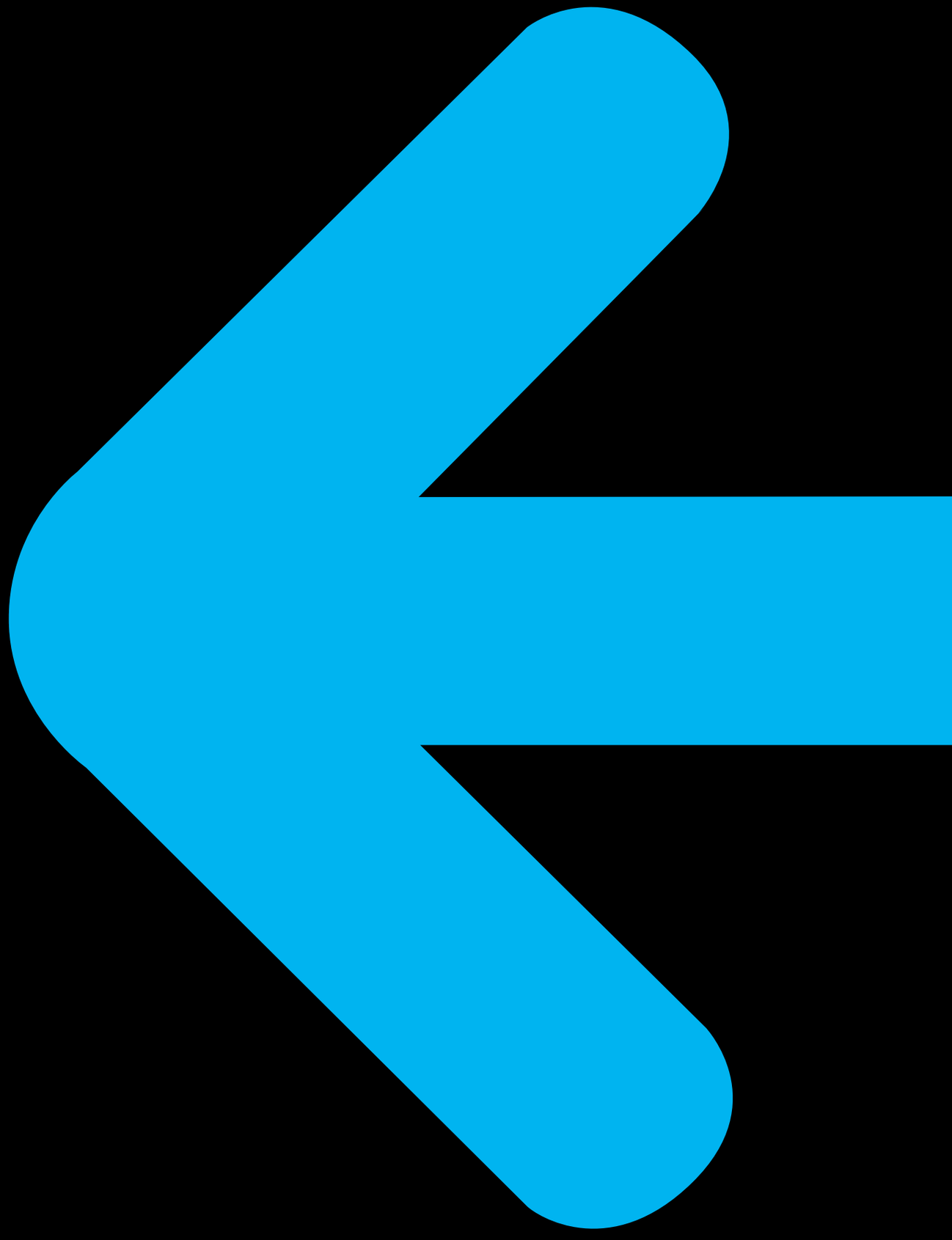
Almost certainly some things will be postponed to 7.1.

vaadin }>

Vaadin 7.1

- Table redesign
- Things postponed from 7.0

more info



vaadin }>

Find out more at

vaadin.com/wiki

vaadin }>

Roots (Windows)

- Creating a basic application
- Creating multi tab applications
- Defining the theme for a Root
- Using URI fragments
- Finding the current Root and Application
- Using URI or parameters or screen size when initializing an application
- Generating dynamic resources based on URI or parameters
- Creating an application that preserves state on refresh
- Creating an application with different features for different clients

vaadin }>

Form & Data binding

- Formatting data in Table
- Creating a TextField for Integer only input using a data source
- Creating a TextField for Integer only input when not using a data source
- Creating your own converter for String <-> MyType conversion
- Changing the default converters for an application
- Creating a master-details view for editing persons
- Auto generating a form based on a bean - Vaadin 6 style Form
- Creating a form using an existing layout
- Using Bean Validation to validate input
- Creating a custom field for editing the address of a person

vaadin }>

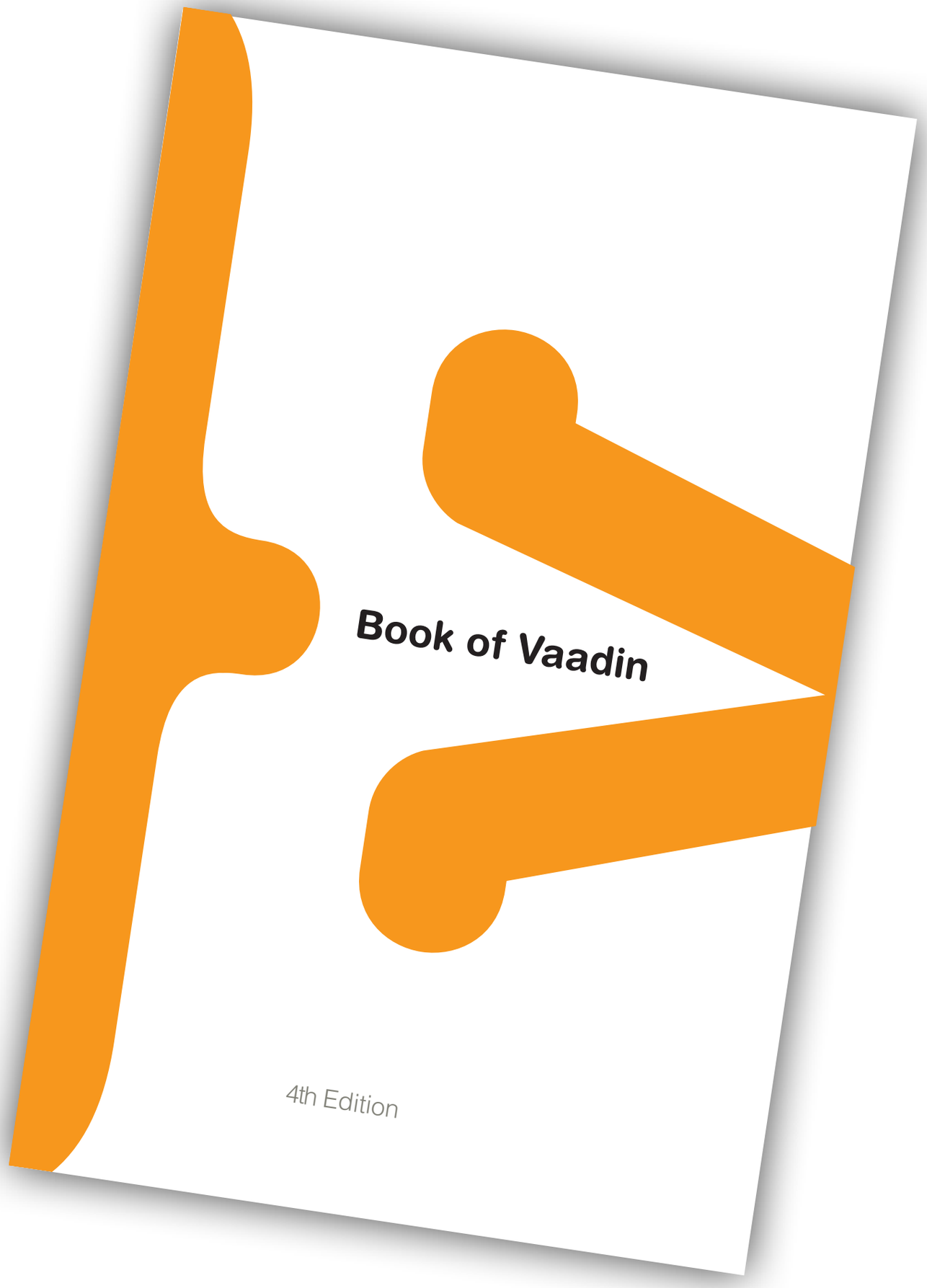
Custom widgets

- Creating a simple component
- Using Shared State for communication
- Sending events from the client to the server using RPC
- Using RPC to send events to the client
- Using Components in the shared state
- Using Resources in the shared state
- Integrating an existing GWT widget
- Creating a simple component container
- Widget styling using only CSS
- Lightweight calculations of widget layout
- Complex widget layouts

vaadin }>

Download for Free

vaadin.com/book



674 pages

DZone Refcardz

brought to you by... vaadin

CONTENTS INCLUDE:

Getting Started with Vaadin

By Marko Grönroos

ABOUT VAADIN

Vaadin is a server-side Ajax web application development framework that allows you to build web applications just like with traditional desktop frameworks, such as AWT or Swing. An application is built from user interface components contained hierarchically in layout components.

In the server-driven model, the application code runs on a server, while the actual user interaction is handled by a client-side engine running in the browser. The client-server communications and any client-side technologies, such as HTML and JavaScript, are invisible to the developer. As the client-side engine runs as JavaScript in the browser, there is no need to install plug-ins. Vaadin is released under the Apache License 2.0.

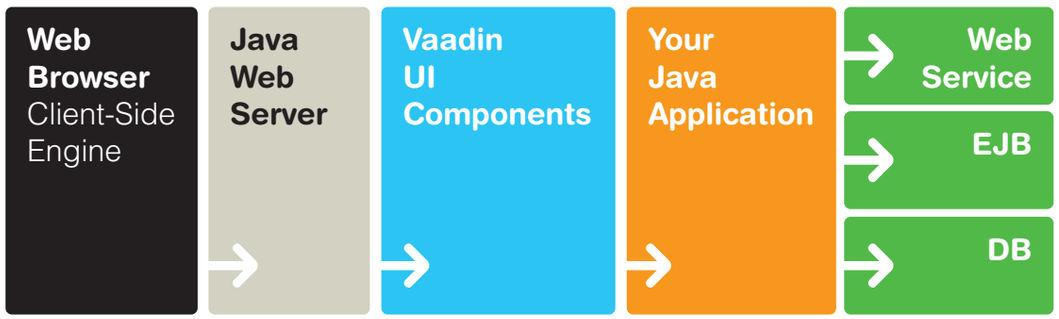


Figure 1: Vaadin Client-Server Architecture

If the built-in selection of components is not enough, you can

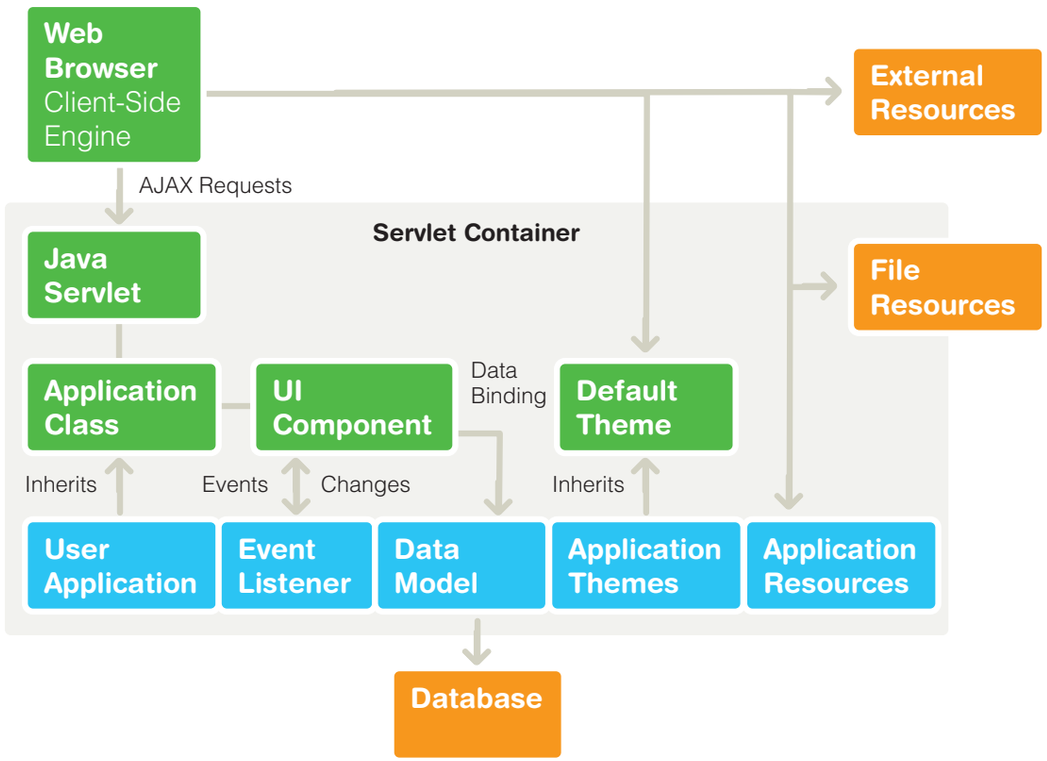


Figure 2: Architecture for Vaadin Applications

Hot Tip You can get a reference to the application object from any component attached to the application with `getApplication()`

Event Listeners

In the event-driven model, user interaction with user interface components triggers server-side events, which you can handle

Questions?
Comments?

joonas@vaadin.com
vaadin.com/joonas
@joonaslehtinen
#vaadin

Slides available on
slideshare