

A man with a beard, wearing a white shirt and a grey vest, is smiling while working on a laptop. He is in a modern office environment with large windows and other people in the background. A water bottle and a glass are on the desk next to him.

**CaptainCasa  
Enterprise Client**

# **XIV. Community Meeting**

# Agenda

- 10:00 Opening / Positioning / 2018 in a Nutshell
- 10:30 Containers and Sizing
- 10:50 SVG Icons
- 11:00 Integration of 3<sup>rd</sup> Party Web Frameworks
- 11:20 SCENARIO „Zürich“
- 11:35 Coffee Break
- 11:50 Style Management
- 12:10 Animations
- 12:20 SCENARIO „Somewhere in the Cloud“
- 12:35 BG Activities
- 12:45 Lunch Break
- 13:30 Docker
- 13:50 Session Management: Cookie <=> URL
- 14:10 SCENARIO „Athens“
- 14:25 Dynamic Page Management
- 14:45 Single Sign On
- 14:55 SCENARIO „Malta“
- 15:10 Coffee Break
- 15:30 CCEE
- 16:00 SCENARIO „Stuttgart“
- 16:15 Project setup
- 16:30 Audit Survival Guide
- 16:45 Road map
- 16:59 Questions & Discussion

# Meetermass








# Where we are

# Roadmap 2018

RISC Client	
Swing	(Maintenance)
FX	(Maintenance)
PageBean Components	
Services	

# Roadmap 2018

RISC Client	 <p>RISC-HTML Client „perfectly“ usable in desktop browser scenarios „nicely“ usable in straight forward mobile scenarios</p>
Swing	(Maintenance)
FX	(Maintenance)
PageBean Components	 <p>PageBean Components                      CCEE</p>
Services	 <p>BG is active!</p>



# CaptainCasa

- Germany

- Björn Müller
- Dr. Ulrich Hirsch

- BG / Evolution IT

- Branimir Ananiev
- Nikolay Nikolov
- Rostislav Gardinov
- Velizar Atanasov
- Atanas Yalandz
- Galin Dobchev
- Alexander Krastev
- Ivan Penkov
- David Kostov
- Rumen Hristov
- Marieta Stoyanova



# RISC Browser Architecture

Complex Instruction Set  
Client  
CISC

Reduced Instruction Set  
Client  
RISC

Framework

#Elements  
#Attributes  
#CSS

Rendering  
„Algorithmic“

Control Libraries  
(.js)  
based  
On  
Nucleus Elements

Rendering „Execution“

Micro Kernel

„2“ El.

# Positioning

## **Powerful HTML Rich Client Framework** for demanding Business Applications

---

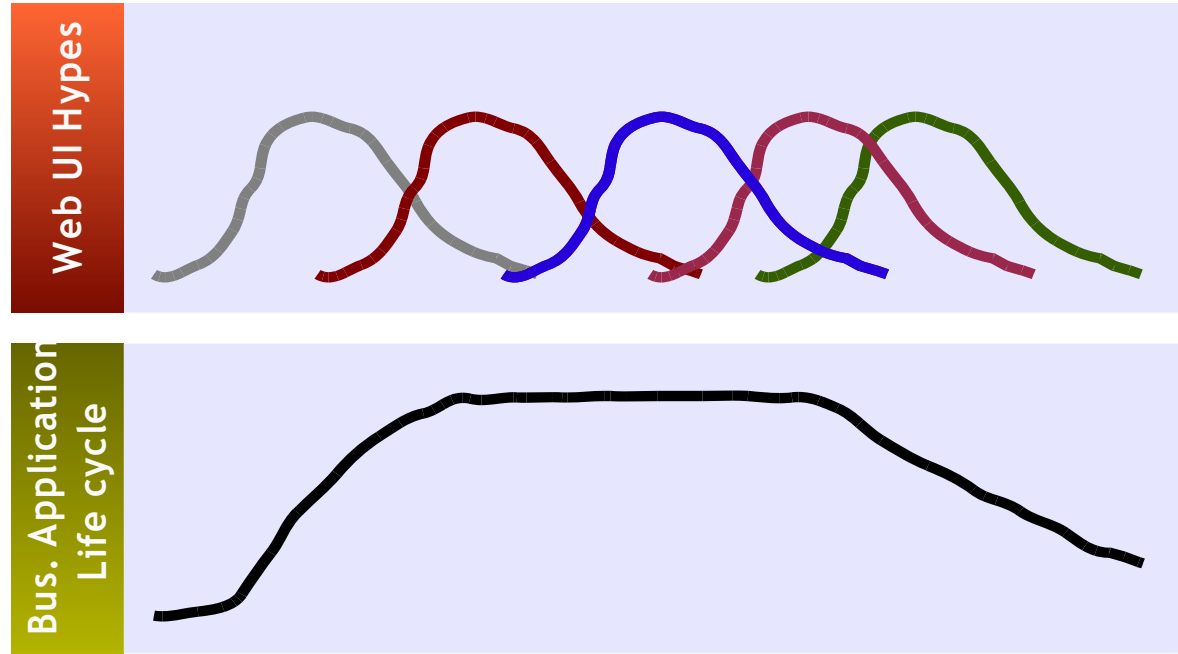
Rock solid, industry-stable  
Cross browser/device  
High performance  
Overcoming HTML restrictions

Simple development, Java-only  
Huge component library  
Based on standards

---

**Using RISC-HTML**

# Still the same old picture



# Outside...

- Direct competitors
  - Vaadin
    - Incompatibility due to change of client framework
      - GWT Google Web Toolkit
      - Polymer
- Indirect competitors
  - AngularJS
  - ReactJS
  - ...
  - Client side development  $\Leftrightarrow$  Server side development

# New Components

# TEXTAREAWITHCOMMENT

Demo    Layout XML    Server side Java Code

```
1 <%%page contentType="text/html"%>
2 <%%page pageEncoding="UTF-8"%>
3
4 <%%taglib prefix="f" uri="http://java.sun.com/jsp/core"%>
5 <%%taglib prefix="h" uri="http://java.sun.com/jsp/html"%>
6
7 <%%taglib prefix="demo" uri="/WEB-INF/democontrols"%>
8 <%%taglib prefix="t" uri="/WEB-INF/ecint"%>
9 <%%taglib prefix="ccaddons" uri="/WEB-INF/ccaddons"%>
10
11
12 <!-- ===== CONTENT BEGIN ===== -->
13 <f:view>
14 <h:form>
15 <f:subview id="workplace_demohelloworldg_sv">
16 <t:rowdemobodypane id="g_1" objectbinding="#{d.demoHelloWorld}" >
17 <t:row id="g_2" >
18 <t:label id="g_3" text="Your Name" width="120" />
19 <t:field id="g_4" clientname="yourName" text="#{d.demoHelloWorld.name}" width="200" userhint="Your Name" />
20 </t:row>
21 <t:rowdistance id="g_5" height="5" />
22 <t:row id="g_6" >
23 <t:rowdistance id="g_7" width="120" />
24 <t:button id="g_8" actionListener="#{d.demoHelloWorld.onHello}" clientname="onHello" text="Hello!" />
25 </t:row>
26 <t:rowdistance id="g_9" height="50" />
27 <t:row id="g_10" >
28 <t:label id="g_11" text="Result" width="120" />
29 <t:field id="g_12" clientname="result" enabled="false" text="#{d.demoHelloWorld.output}" width="100%" />
```

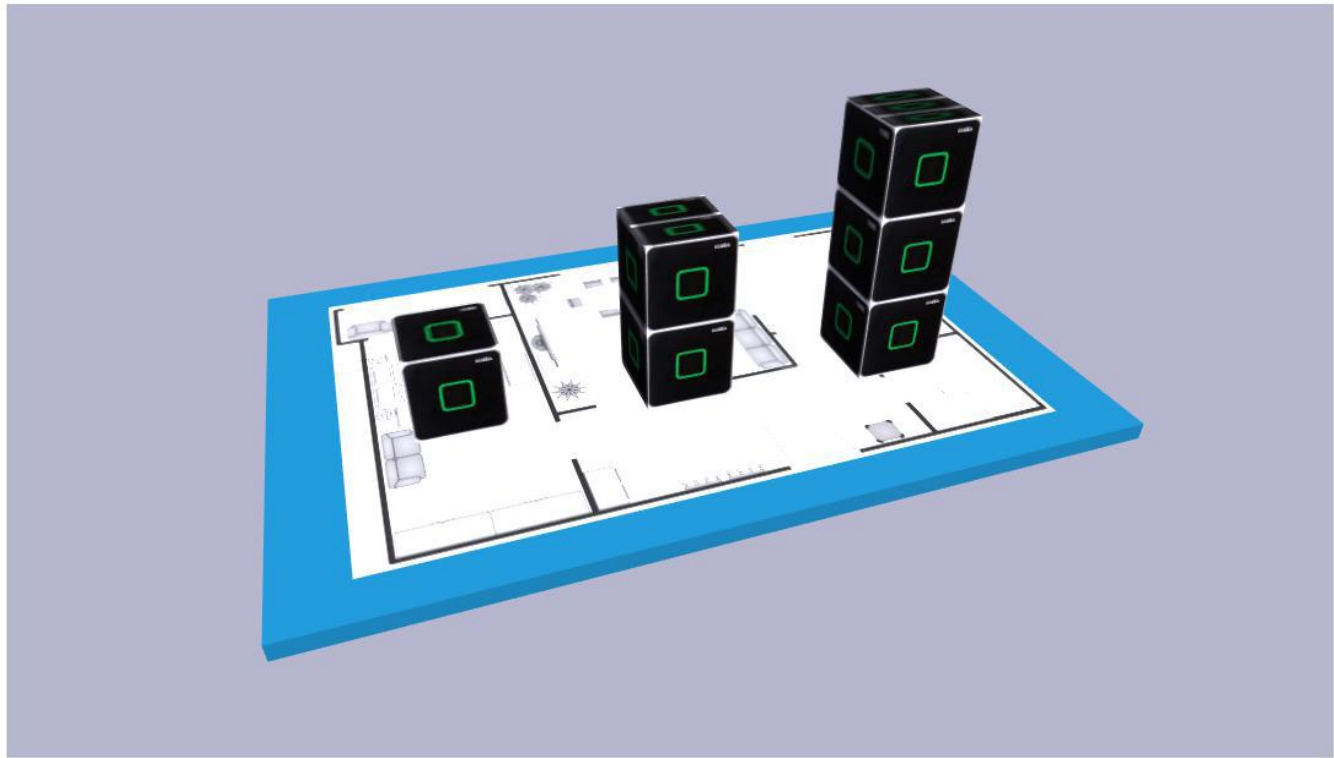
The component TEXTAREAWITHCOMMENT shows a normal text area on the right.

On the left there is a parallel text area which renders a comment. The comment typically is some numbering, but may also be something different.

Default numbering    Own numbering

# SCENEJS

Demo   Layout XML   Server side Java Code



In this example the Scenejs-graph is built up dynamically using Map-objects and using the Google GSON-library for creating JSON strings.

In addition name-nodes are passed into the graph, which can be selected by the user. So if clicking one of the cubes the corresponding information is sent back as event to the server side processing.



# SVG „everywhere“



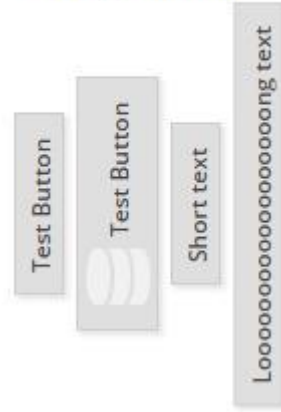
100% rendering



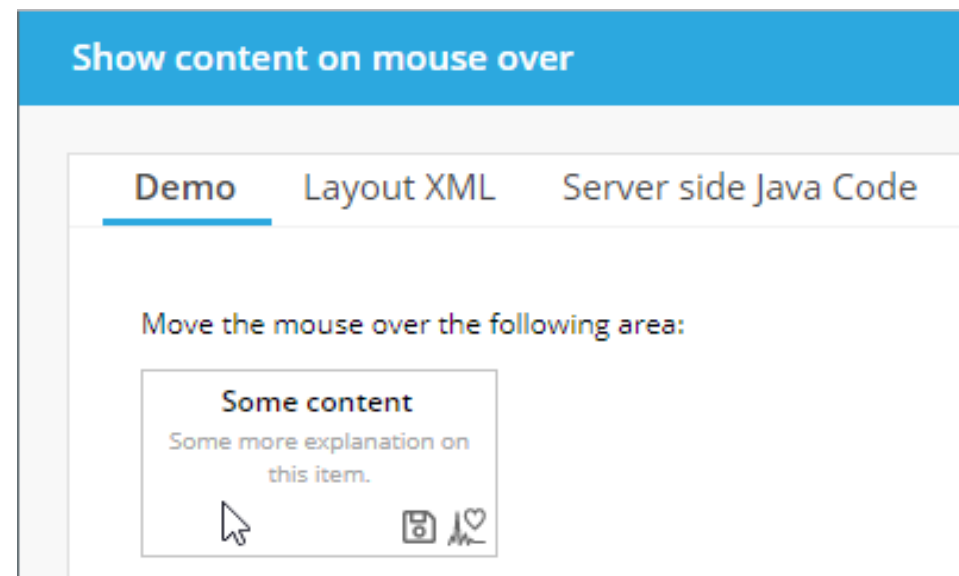
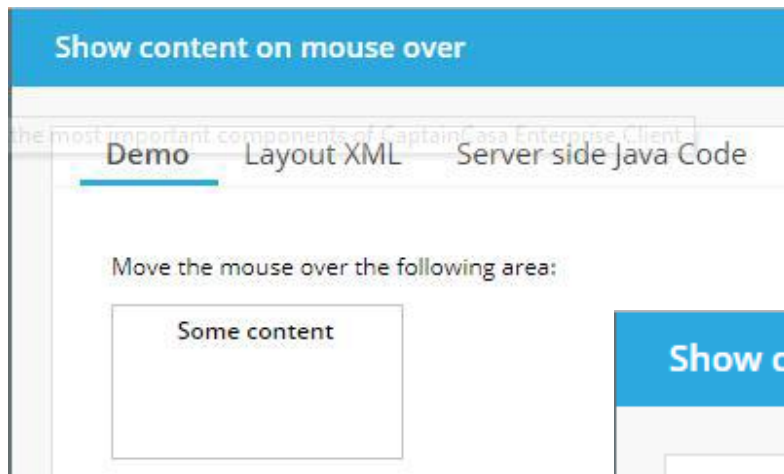
150% rendering

# BUTTONVERTICAL

Sized by content



# OVERLAYAREAITEM- SHOWONAREAMOUSEOVERONLY

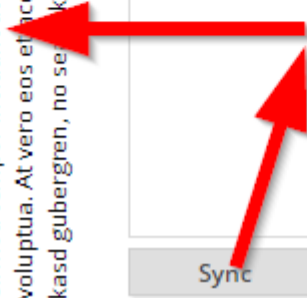


# TEXTPANEVERTICAL

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Sync



# Explicit Popup Styling

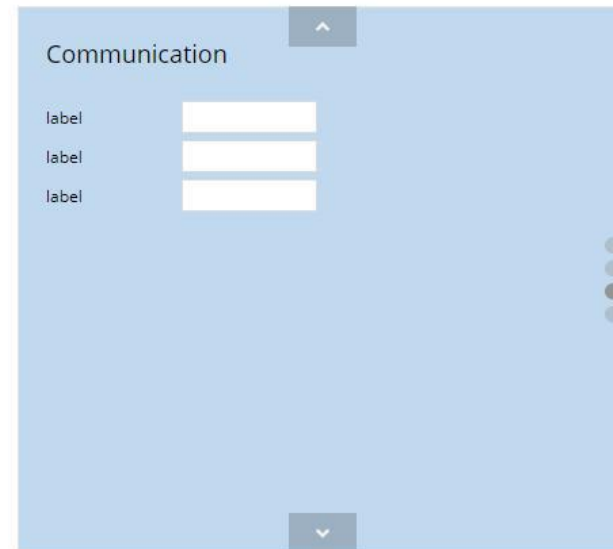
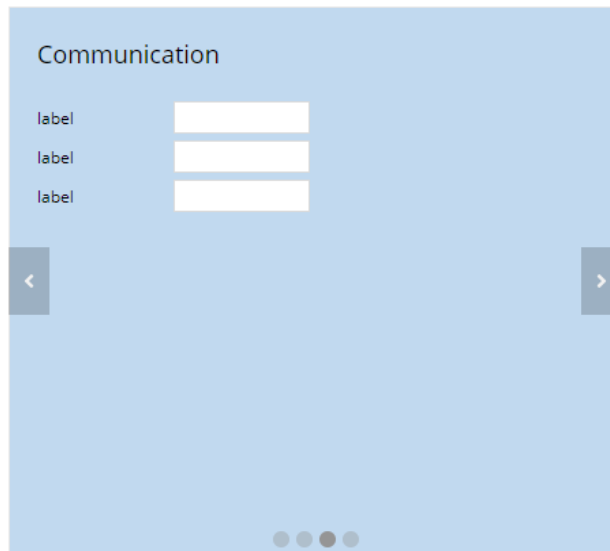
## Modal/ModelessPopup.setStyleSeq(...)

The screenshot displays the CaptainCasa IDE interface. On the left, a sidebar contains a 'New Functions' tree with 'Own style class for popup dialogs' selected. The main workspace shows a 'Demo' window with tabs for 'Layout XML' and 'Server side Java Code'. The 'Demo' tab contains a 'Default style' button and a 'Style "transparent"' button. A yellow callout box highlights the text: 'Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.' A 'Close' button is visible at the bottom right of the callout.

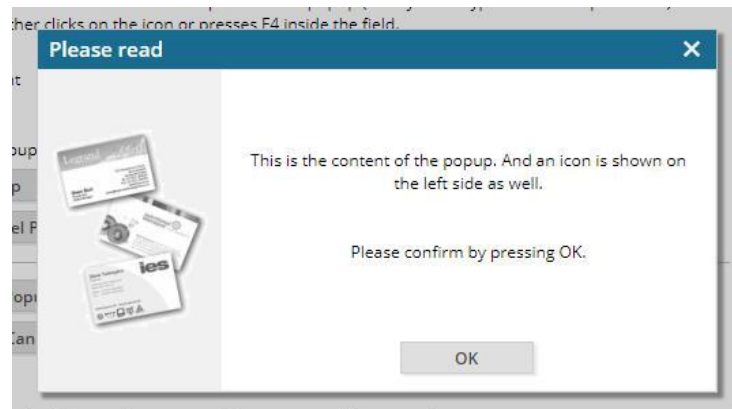
On the right, a text box explains: 'By default the popup dialog is styled using the style class "riscdialog\_popup" - which in the default style is defined as area with white background and with a certain shading. But you can also extend this style class by own style classes - and then pass the name of the style class by calling method Modal/ModelessPopup.setStyleSeq(..)'.

Below this, a smaller 'Demo' window shows a button labeled 'Open Popup'. Clicking it opens a modal popup dialog with a white background and a grey border. The dialog contains two rows of 'Some label' and 'Some field' text, and a 'Close' button at the bottom right.

# SHIFTCONTAINER++



# Flat Design of default Dialogs



# SHADOW


The screenshot displays the CaptainCasa Enterprise Client RISC interface. On the left, a vertical sidebar contains a 'Dashboard' section with 'Components' and 'New/Updated Functions'. The 'New/Updated Functions' list includes items like 'Nov 05th, 2018 - new' with sub-items 'Applying shadows', 'Oct 22nd, 2018 - new' with 'Chinese touch keyboard', and others. Below this are 'Containers', 'Grids & Trees', 'General', and 'Add-ons'. The main area is titled 'Applying shadows' and features a 'Demo' tab. The demo shows a UI with a text field 'Some field', a button 'Some button', and a title 'Some title', all with shadows. A text box on the right explains the SHADOW attribute: 'All graphical components provide a SHADOW attribute. The definition of the value internally consists out of 3 definitions: 1. the horizontal offset of the shadow 2. the vertical offset of the shadows 3. the blurring around the border of the shadow. If defining "0" then the border of the shadow is "exact!". If defining "4" then there is some blurring 2 with a size of 2 pixels on each side of the border line. The shadow definition automatically updates the component's size: a margin is arranged around the component.'



# QR CODE Scanner

Demo   Layout XML   Server side Java Code

Place a QR Code in front of your camera...



Protocol:  
QRCode was scanned: <http://www.example.com>, Tue Nov 20 06:10:49 CET 2018  
QRCode was scanned: <http://www.example.com>, Tue Nov 20 06:10:51 CET 2018  
QRCode was scanned: <http://www.example.com>, Tue Nov 20 06:10:52 CET 2018  
QRCode was scanned: <http://www.example.com>, Tue Nov 20 06:10:53 CET 2018  
QRCode was scanned: <http://www.example.com>, Tue Nov 20 06:10:54 CET 2018  
QRCode was scanned: <http://www.example.com>, Tue Nov 20 06:10:55 CET 2018

# GEOLOCATION

## GPS Sensor Information

Accuracy	2.820	Protocol:
Longitude	8,8006656	Event: OK at Mon Dec 03 09:57:08 CET 2018
Latitude	49,3780992	



# CAMERA

The screenshot shows a web browser window with the URL `localhost:50000/demos/workplace.workplaceRisc.risc`. The browser's taskbar includes several open applications: DEMO8080, EDITOR8080, DEMO50000, EDITOR51000, BPMN Analyser, Leitstand, HelloWorld, and Neuer Tab. The application interface is divided into a left sidebar and a main content area.

**Left Sidebar:**

- Components**
- Containers**
  - Structuring page content
  - Panes
    - Foldable / Titled Pane
    - Tab Pane
    - Tab Pane (server side)
    - Sizeable Pane
  - Applying shadows
  - Background Painting
  - Container sizing
  - Layering
  - Adaptive Containers
  - Runtime update of size
  - Alignment in Rows
  - Scroll Pane - Extended Functions
  - Split Pane - Extended Functions
  - Tiled Area
  - Transfer of actual Size
  - Focus Observation
  - Cascade rendered=false
  - Shift Container
  - Hideable Container
  - Scale Pane
  - Slide Container
  - Blockable Area
- Grids & Trees**
- General**
- Add-ons**

**Main Content Area:**


The main area is titled "Camera - take photo" and contains a "Demo" tab. It features a large video feed of a man in a white shirt. Below the video is a "Captured image" section showing a smaller version of the same image. To the right of the video, there is explanatory text:

The CAMERA component shows what is currently recorded by the web camera. By pressing the "snap"-button, the current image is sent to the server side in its original camera resolution.

The CAMERA component requires some explicit activation by the user for security reasons. It is NOT available on Internet Explorer 10/11.

The bottom of the browser window shows several open tabs: "Hello World!", "Basic Controls", "Simple Grid", "Structuring page content", and "Camera - take photo".

# ROWMATRIXAREA

<p>Hello</p> <p>world</p>	<p>First name</p> <input type="text"/> <p>Last Name</p> <input type="text"/>	<p>Some button</p>	
<p>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum..Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.</p>	<p><u>First</u> Second <span>☰</span></p> <p>First name</p> <input type="text"/> <p>Last Name</p> <input type="text"/>	<p>Hello</p> <p>world</p>	

# TOUCHKEYBOARD

The screenshot displays a software interface with a sidebar on the left and a main content area on the right. The sidebar contains a 'Components' menu with the following items:

- Some simple examples
- Straight Controls
- Scheduling
- Animation & Graphics & Sound
- Touch Screen
  - Touch field
  - Numeric field
  - Field with touch popup
  - Point of Sales
  - Touch Keyboard Layouts
  - Touch Grid Navigation
  - Signature
  - Keyboard Scanner
  - Chinese keyboard
  - QR Code Scanner (JSQR)
  - Stable virtual keyboard
- Reporting, PDF, SVG
- Mobile Periphery

The main content area is titled 'Stable virtual keyboard' and features a 'Demo' tab. The demo shows a form with the following fields:

- First Name:
- Last Name:
- Shoe Size:
- Married:
- Height:
- Department:

Below the form is an 'OK' button. To the right of the form is a text box containing the following text:

The TOUCHVIRUTALKEYBOARD is a software keyboard component that is not bound to a specific field/component but that operates indenpently as a vritual keyboard.

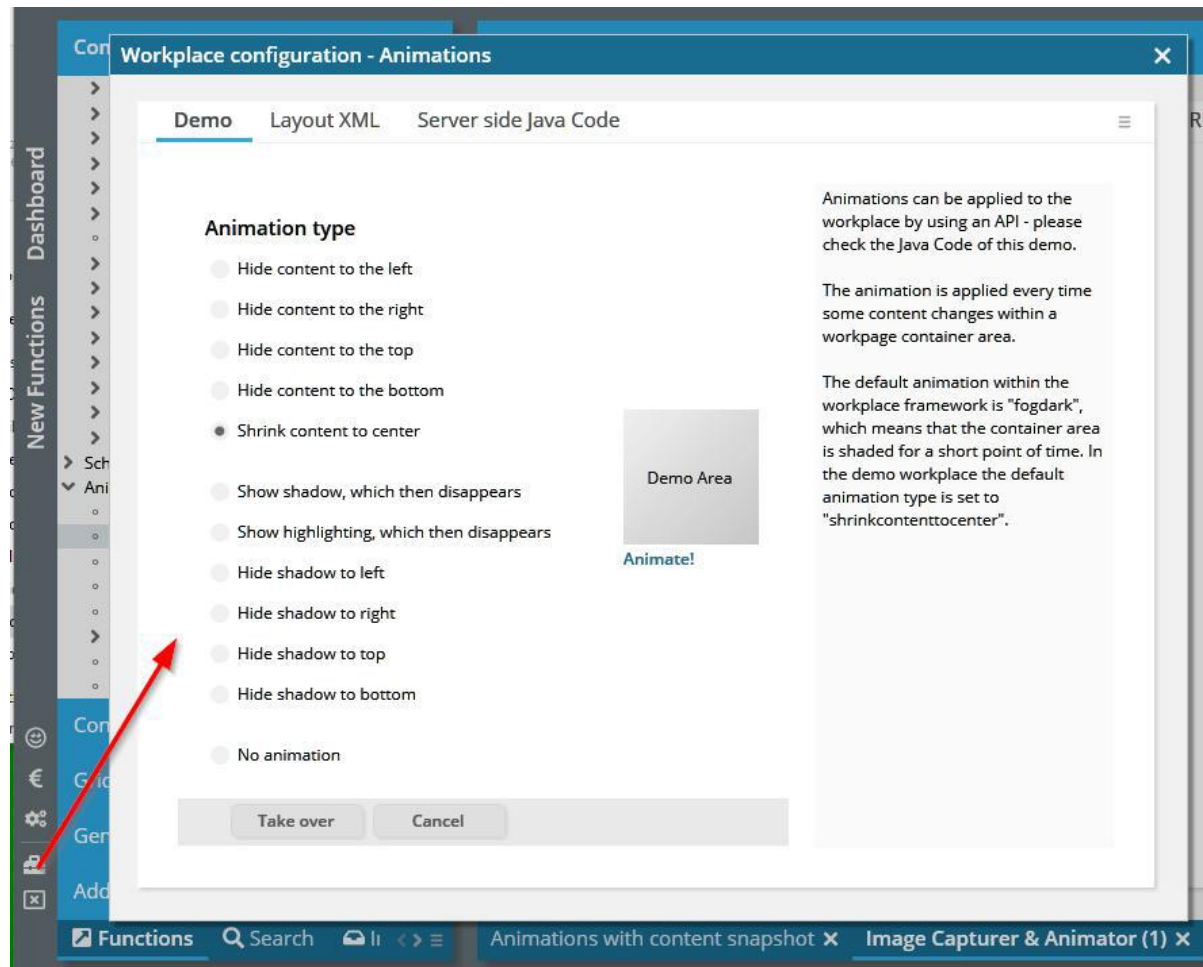
At the bottom of the interface is a virtual keyboard with the following layout:

- Row 1: 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, DEL, C, <<, >>
- Row 2: @, q, w, e, r, t, z, u, i, o, p, ü, #\$\$
- Row 3: a, s, d, f, g, h, j, k, l, ö, ä, ABC
- Row 4: Caps, y, x, c, v, b, n, m, ., -, -
- Row 5: Shift, Space, OK

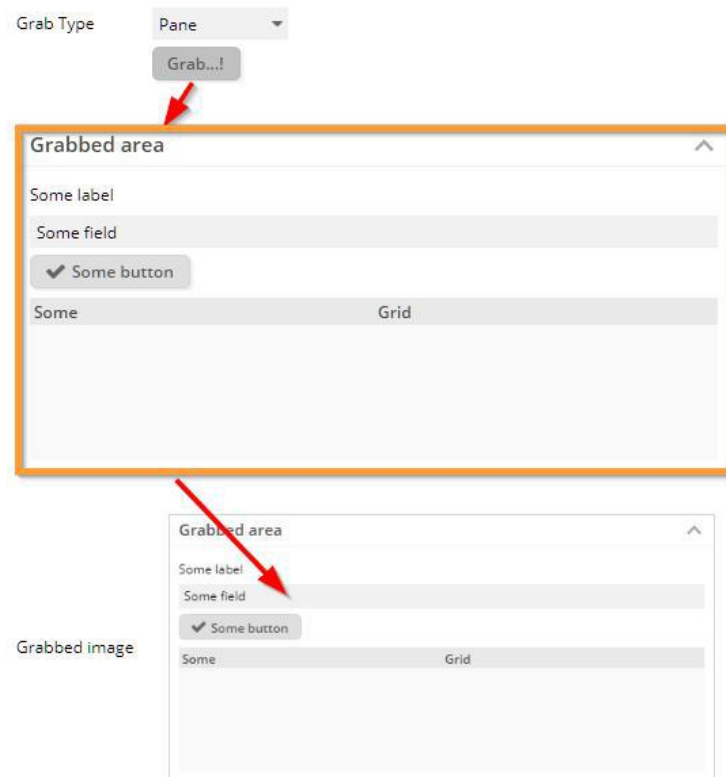
The bottom of the interface shows a navigation bar with the following items:

- Mobile show case x
- Rolling container area x
- Stable virtual keyboard x

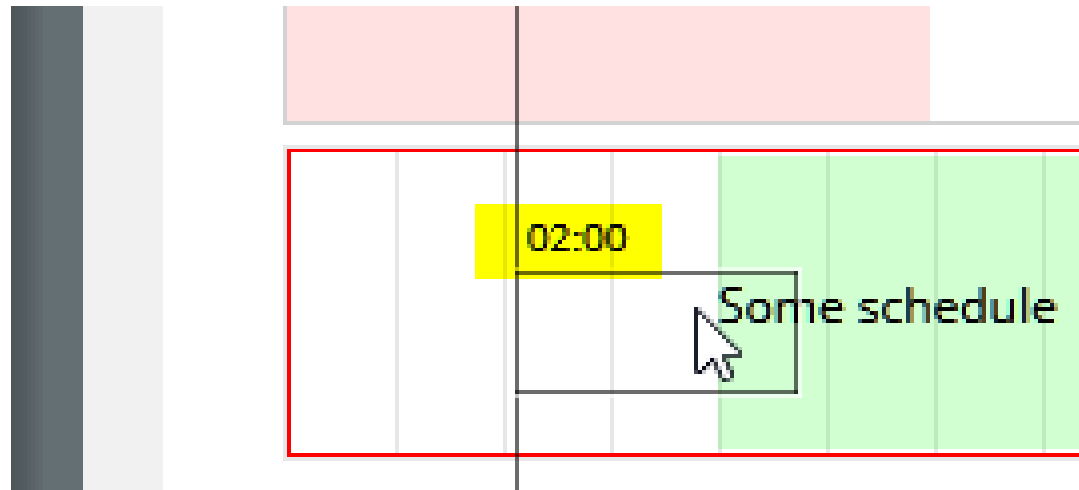
# Better Animations



# SCREENGRABBER



# DROPRASTERTEXT





# POOLEDBROWSER

The screenshot displays the CaptainCasa Enterprise Client RISC interface. On the left is a navigation sidebar with sections: Components, Containers, Grids & Trees, General, New Functions, and Add-ons. The 'New Functions' section is expanded, listing various features like 'Popup Dialogs', 'Includes', 'Page Beans', 'Blockable Area', 'Dynamic Layout', 'Asynchronous Processing', 'Adapter Binding', 'Style and Language Management', 'Client Clipboard', 'Tool Tips / User Hints', 'General Issues', 'Server Side Value Binding', 'Server side JFreeChart', 'Workplace Management', 'Client Integration', and 'RISC Client Integration' with sub-items like 'Camera / take photo', 'Geo location', 'Insert JavaScript', 'Insert HTML', 'Keyboard Scanner', 'Combination: Scanner <=> Field input', 'Subpage', 'Iframe inclusion with JS interface', 'Playing videos', 'Playing Youtube movies', and 'Pooled browser'. The main content area is titled 'Pooled browser' and shows the 'Enterprise Client Community' forum page. The forum header includes the 'CaptainCasa' logo, search, recent topics, member listing, and back to home page links, along with 'Register / Login' options. The forum title is 'All Issues XML' and the page shows a list of forum topics with columns for Topic, Answers, Author, Views, and Last message. The topics listed are: 'IE 11 - ActiveX', 'Exceptions on CC-SDM 20110406 when using configuration', 'Doubts on using functions', 'Cannot use CC-SDM 20100802 on Enterprise Client 4', 'SDM acces to other relational database', 'CC-SDM version 20100802', and 'First posting...'. Below the list is a 'Go to:' dropdown menu and a 'Go' button. At the bottom, there are status indicators for 'New messages', 'No new messages', 'Announce', 'Sticky', and 'Forum Leaders:'. The footer of the forum page reads 'Powered by JForum 2.1.6 © JForum Team'.

Topic	Answers	Author	Views	Last message
IE 11 - ActiveX	1	EOsmanliu	3487	23/09/2014 16:53:52 CaptainCasa →
Exceptions on CC-SDM 20110406 when using configuration	7	cvieira	3711	07/04/2011 12:23:35 cvieira →
Doubts on using functions	0	cvieira	1253	07/04/2011 12:04:35 cvieira →
Cannot use CC-SDM 20100802 on Enterprise Client 4	8	cvieira	2255	06/04/2011 11:24:42 CaptainCasa →
SDM acces to other relational database	1	lkuetem	1570	07/02/2011 17:27:02 CaptainCasa →
CC-SDM version 20100802	0	CaptainCasa	1152	03/08/2010 06:39:02 CaptainCasa →
First posting...	0	Admin	1438	26/07/2010 13:41:29 Admin →

# MultiValueSelection

The image shows a screenshot of a 'MultiValueSelection' widget. It features a search bar at the top with a magnifying glass icon. Below the search bar is a list of six items, each with a square selection checkbox on the left and a small upward or downward arrow on the right. The items are: Bammental, Bayrischzell, Sofia, Karlsruhe, München, and Stuttgart. At the bottom of the widget, there are two buttons: 'Alle selektieren' (Select all) and 'Selektion löschen' (Clear selection).

Item	Selected
Bammental	<input type="checkbox"/>
Bayrischzell	<input type="checkbox"/>
Sofia	<input type="checkbox"/>
Karlsruhe	<input type="checkbox"/>
München	<input type="checkbox"/>
Stuttgart	<input type="checkbox"/>

Buttons:

# DateRangeSelection

Date range



The date-range-selector-button (class

Jahr **Monat** Woche

01.03.2019 - 31.07.2021

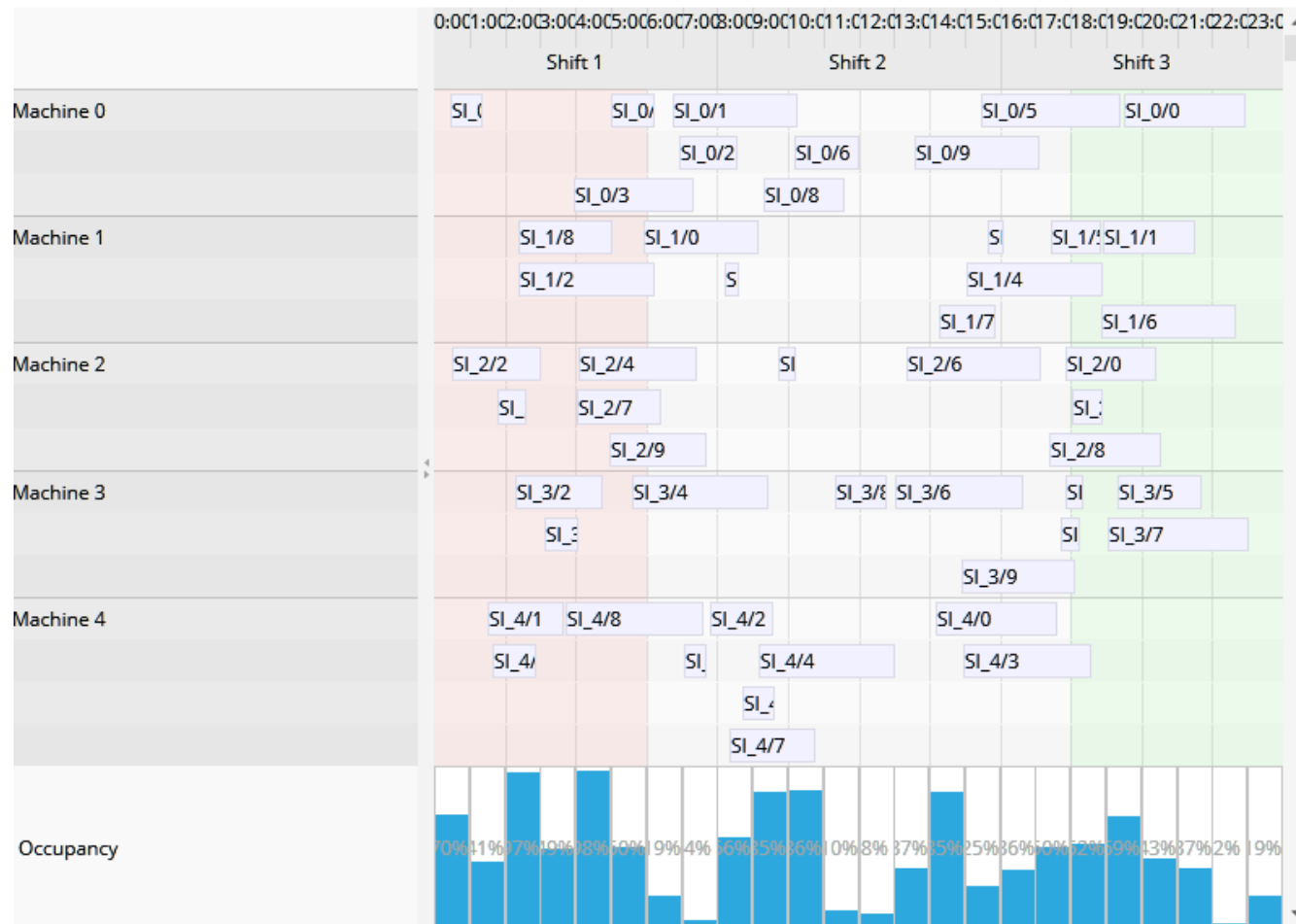
	Jan	Feb	Mär	Apr	Mai	Jun	Jul	Aug	Sep	Okt	Nov	Dez
2015	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2016	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2017	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2018	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2019	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2020	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2021	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2022	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2023	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2024	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2025	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←
2026	→ 1 ←	→ 2 ←	→ 3 ←	→ 4 ←	→ 5 ←	→ 6 ←	→ 7 ←	→ 8 ←	→ 9 ←	→ 10 ←	→ 11 ←	→ 12 ←

✓ OK

# AttributeAssignment

- ▼  Color
  - Red
  - Red
  - Blue
- ▼  Font size
  - 12 - default
  - 16 - headline
  - 20 - title
  - 30 - special
  - 50 - special
- ▼  Font family
  - Arial
  - Trebuchet MS

# MultiLineSchedule

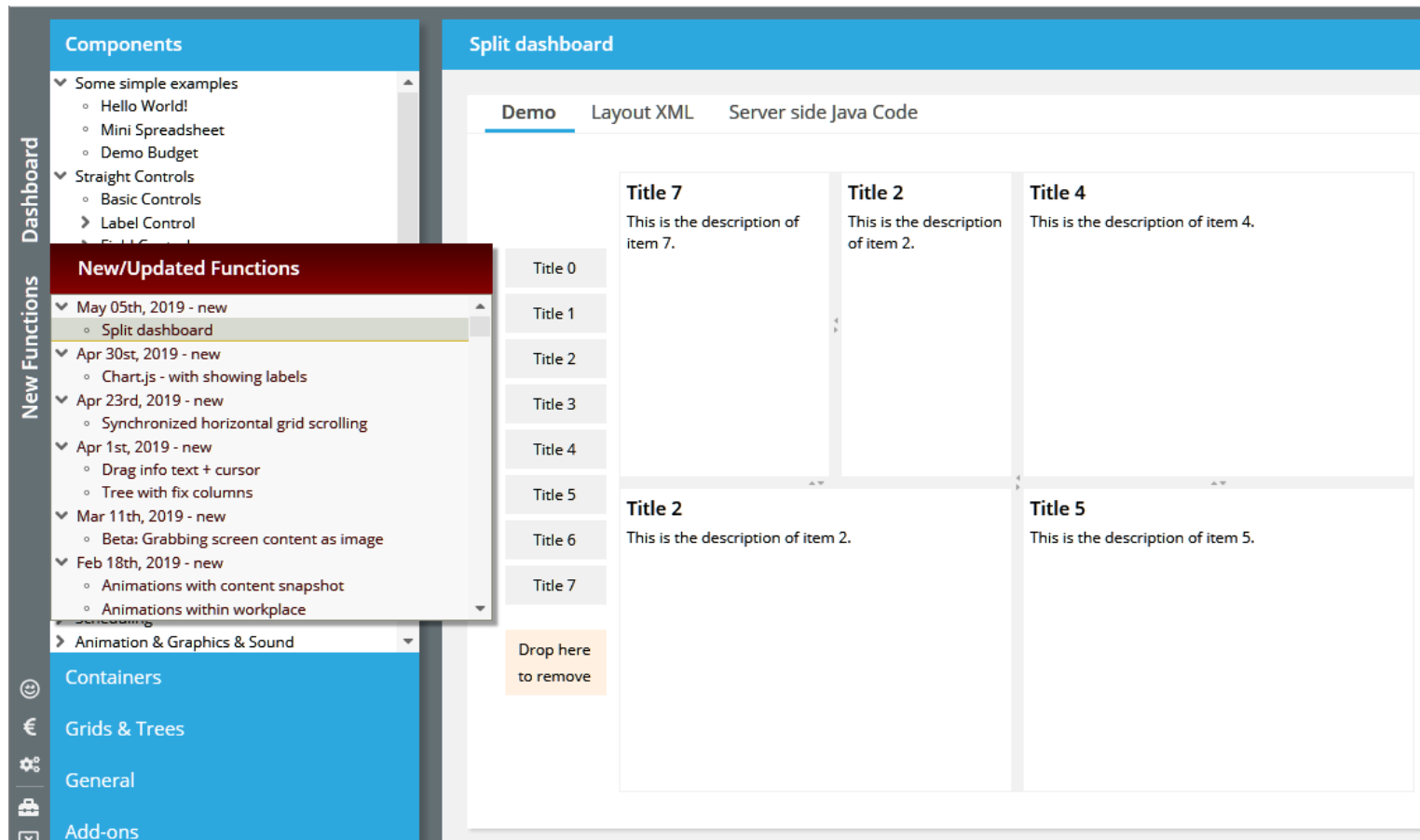


# FULLSCREENBUTTON

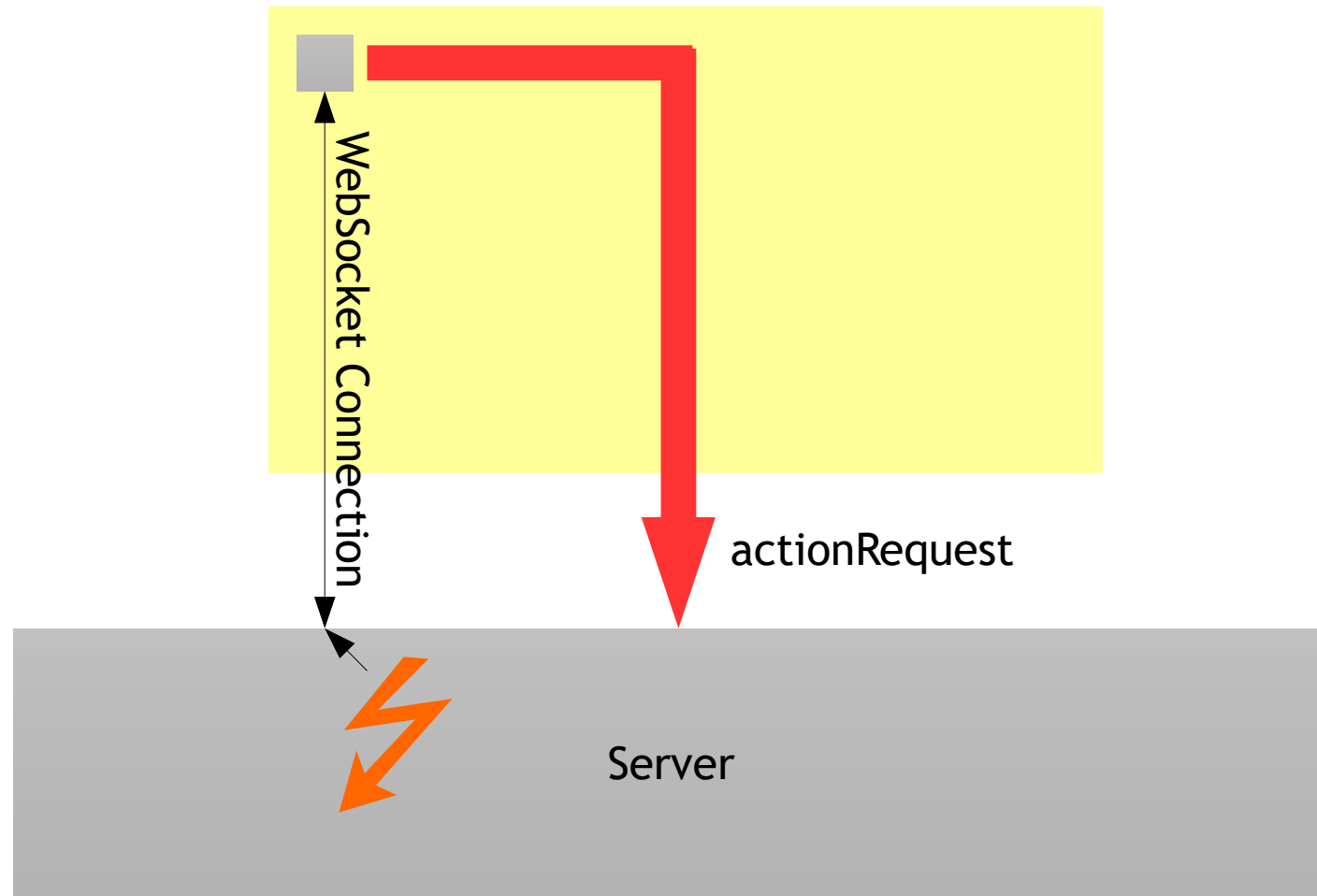
Demo Layout XI

Full screen!

# SPLITDASHBOARD



# WEBSOCKETPOLLING





# Project Setup

# Project structure

CaptainCasa

Maven

src  
webcontent

src  
main  
java  
resources  
webapp  
test

webcontentbuild  
WEB-INF  
classes

target  
classes  
<projectName>  
<projectName>.war

webcontentcc

webcontentcc

pom.xml

# CaptainCasa tool setup => .ccproject

CaptainCasa

src  
webcontent  
webcontentbuild  
WEB-INF  
classes  
webcontentcc

tomcat  
webapps  
cm2019

(always)

RISC

```
93 <project>
  .. webcontentdirectory="${project}/webcontent"
  .. javasourcedirectory="${project}/src"
  .. javaclassdirectory="${project}/webcontentbuild/WEB-INF/classes"
  .. webappaddonsdirectory="${project}/webcontentcc"
99 ..
10 .. webcontentdeploydirectory="c:/bmu_jtc/EnterpriseClient/server/tomcat/webapps/cm2019"
10 .. webcontextroot="cm2019"
10 .. webhostport="localhost:50000"
10 .. copywebapp="true"
10 .. reloadwebapp="true"
10 >
10 .. <deploycopyinfo fromdir="${project}/webcontentbuild" todir="${projectdeploy}"/>
10 .. <deploycopyinfo fromdir="${project}/webcontentcc" todir="${projectdeploy}" skipduringhotdeploy="true"/>
109 </project>
```

Capta

# CaptainCasa tool setup => .ccproject

Maven

src  
  main  
    java  
    resources  
    webapp  
  test

target  
  classes  
  <projectName>  
  <projectName>

webcontentcc

pom.xml

tomcat  
  webapps  
  cm2019

```
1 <project
2 ..... managedbycctoolset="false"
3 ..... webcontentdirectory="${project}/src/main/webapp"
4 ..... javasourcedirectory="${project}/src/main/java"
5 ..... javaclassdirectory="${project}/target/classes"
6 .....
7 ..... webappaddondirectory="${project}/webcontentcc"
8 .....
9 ..... webcontentdeploydirectory="${deploytomcatwebappsdir}/${projectname}"
10 ..... webcontextroot="${projectname}"
11 ..... webhostport="localhost:${deploytomcatport}"
12 ..... copywebapp="true"
13 ..... reloadwebapp="true"
14 ..... >
15
16 ..... <deploycopyinfo fromdir="${project}/target/${projectname}"
17 ..... ..... todir="${projectdeploy}">
18 ..... </deploycopyinfo>
19
20 </project>
21
```



# Creating a Maven project

- Create project in Eclipse
  - Maven-build one time!
  - (wecontentcc is downloaded)
- Import project into CaptainCasa toolset
- Decision during development
  - Quick deploy
    - Hot deploy!
  - Heavy deploy
    - Maven-build
    - Reload

# Decoupling Toolset $\Leftrightarrow$ Application

- Old
  - The toolset should be the same version as the application
- „New“
  - Decoupling!
  - Any scenario allowed
    - Toolset > Application
    - Toolset < Application

# Gradle...

- Whom of us is using Gradle?
  - ...could we share our knowledge?
  - Minimum goal: Gradle-Setup-Guide



# Why not use both together...?

- Demo
  - One separate Maven project to resolve dependencies
  - Copy \*.jar file over into ANT-managed projects
- FYI
  - Possibility to also add Maven-jar-resolution as ANT-taks, but I never tested this...

# Dynamic Content Processing

# (ROW)DYNAMICCONTENT

- Before
  - Static data rendering was simple
  - Dynamic data rendering was complex
  - Event processing was complex
    - You always had to build up „shadow“ data lists in order to be reachable by expression
- Now
  - Static data rendering is simple
  - Dynamic data rendering is simple
  - Event processing is simple

# XXXXNode

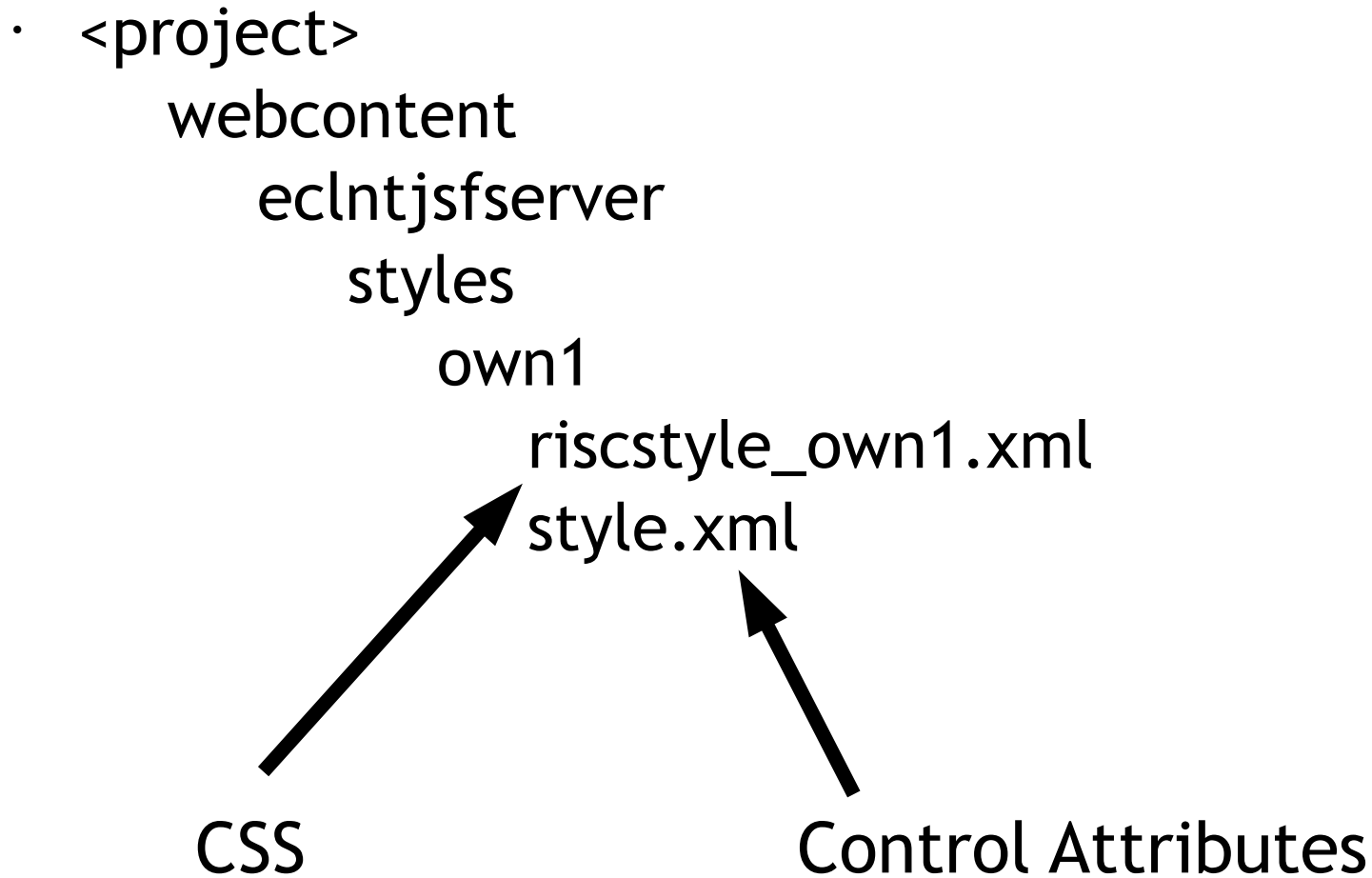
- Old
  - setXxxx / getXxxx
- New
  - bindXxxx(IDynamicContentObjectBinding)
- IDynamicContentBindung
  - IPageBean
  - FIXGRIDBinding (List/Tree)
  - \*Binding classes
  - Simple values: use wrapper
    - \*Delegation (e.g. StringDelegation)

# Do not forget about the REPEAT component!

- !!! - ?

# Style Management

# No change in structure...



# But...

- No more compilation of „.css“
  - „.css“ are dynamically generated at runtime
- No more hacking of XML
  - Use tool! (!!!)



# The „special things“

- CaptainCasa needs to know about
  - sizing
  - coloring
- Sizing
  - inset  
margin  
border
  - font
- Coloring
  - background

# Background color management

- Problem: no overlaying is possible via CSS!
- But: overlaying is heavily required
  - Grid rollover
  - Grid selection
  - Disabling
- The HTML „background“-style provides overlaying!
  - `style="background: linear-gradient(rgba(51, 96, 127, 0.176), rgba(51, 96, 127, 0.176)), linear-gradient(rgba(0, 0, 0, 0.02), rgba(0, 0, 0, 0.02)), linear-gradient(rgba(255, 0, 0, 0.07), rgba(255, 0, 0, 0.07)), rgb(240, 156, 164)"`

# Consequence

- CaptainCasa uses background overlaying for all background operations
  - CaptainCasa needs to know the „base background“ of the component in order to add
    - rollover/selection
    - highlighting
    - disabling
- RISC values
  - „\_backgroundModifierFocus\*“

# Background modifiers

\_backgroundModifierFormatError  
\_backgroundModifierFormatMandatory

\_backgroundModifierFocus  
\_backgroundModifierDisabled  
\_backgroundModifierFocusDisabled

\_backgroundModifierHover  
\_backgroundModifierActive

\_backgroundModifierGridSelection  
\_backgroundModifierGridOdd  
\_backgroundModifierGridEven  
\_backgroundModifierGridRollover

\_backgroundModifierPaintAreaItemSelection

# And some other special things...

- Some components outsource configure-able values into RISC style values

```
function RISCSingleTreeNode(asPrototype)
{
  this.getAccessibilityControlTitle_parent_singletreenode = this.getAccessibilityControlTitle;
  this.reactOnRISCEvent_parent = this.reactOnRISCEvent;
  this.setStyleClass_parent_singletreenode = this.setStyleClass;
  this.getComponentText = function() { return this.getTreeText(); };
  this.init_RISCSingleTreeNode = function()
  {
    this.init_RISCComponent();
    this.m_level = -1;
    this.m_status = -1;
    this.m_indentPerLevel = 20;
    this.m_indentPerLevelExplicitlySet = false;
    this.m_imageSize = 16;
    this.setFocusable(true);
    this.m_statusElement = new RISCElementImage();
    this.m_textElement = new RISCElement();
    this.addSubElement(this.m_statusElement);
    this.addSubElement(this.m_textElement);
    this.m_toggleImageCursor = RISCSTYLEAccess.p(RISCSTYLEPARAM_toggleImageCursor,this,"pointer");
    this.m_toggleImageCursorEndNode = RISCSTYLEAccess.p(RISCSTYLEPARAM_toggleImageCursorEndNode,this,"default");
    // events
    if (RISCUtil.isDesktop())
    {
      this.m_statusElement.setClickCallback(this);
      this.setKeyCallback(this);
    }
    else
    {
      RISCSTYLEPARAM_toggleImageCursor = "_toggleImageCursor";
      RISCSTYLEPARAM_toggleImageCursorEndNode = "_toggleImageCursorEndNode";
    }
    this.m_content = null;
    this.setAccessibilityControlName("SingleTreeNode");
    this.setStyleClass("riscsingletreenode");
  };
}
```

# Control Attribute Styling

- ...no changes! - And still very useful
  - Control variants
  - Maintenance of server side control attributes
    - FIXGRID-SBVISIBLEAMOUNT

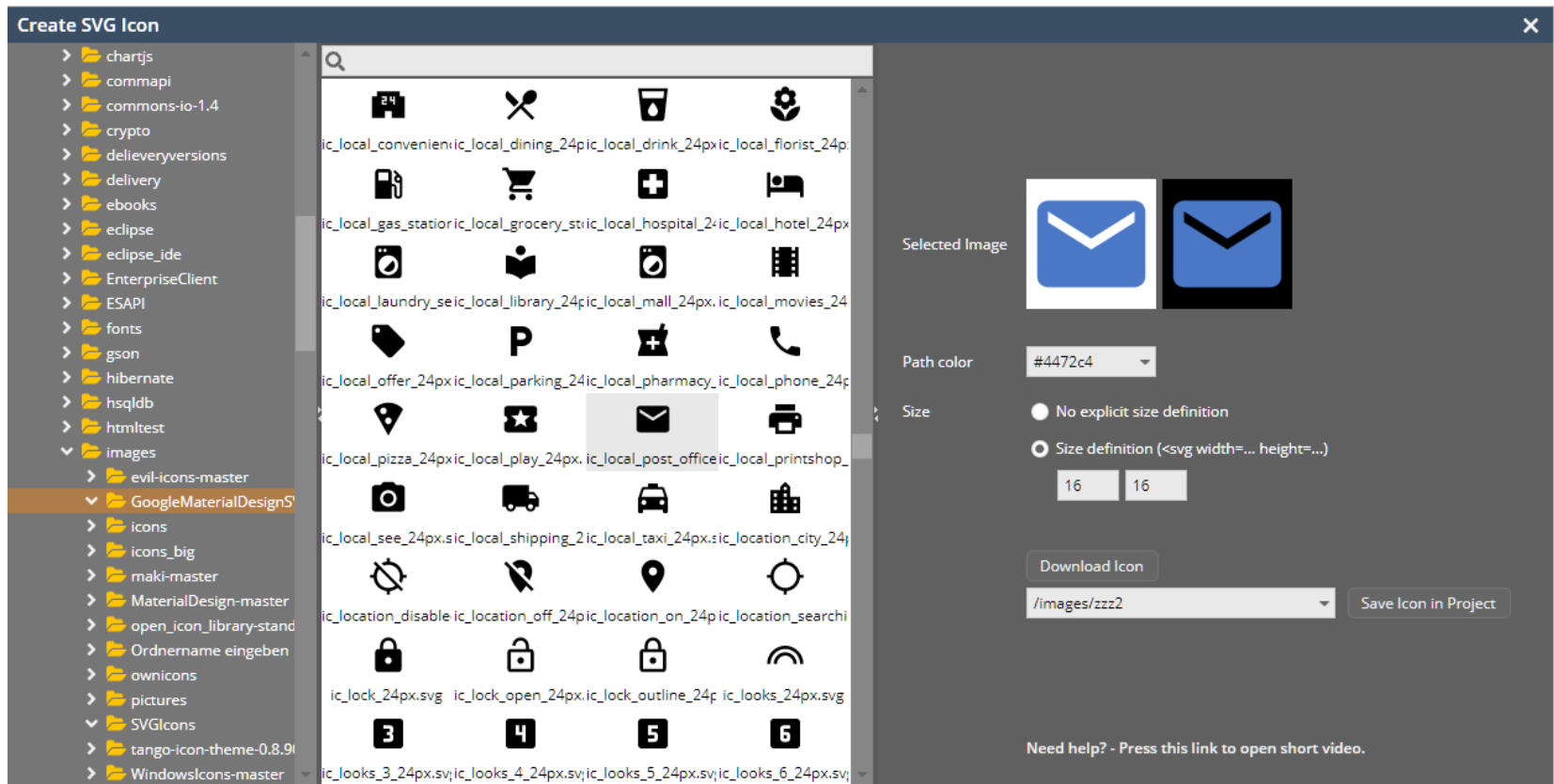
# SVG Icons

# SVG Icons are the (?) way to go

- Advantages
  - Scalable
  - Adapt-able (colors)
  - Huge number of libraries
  - Useable in styles
- (Little) Disadvantages
  - Sizing
  - Adapt-able - but requires some SVG update
- **Use SVG Icons!**
  - Do not use bitmap icons!



# SVG Icon Tool



# Dynamic SVG Adaptation

- Original image
    - /images/camera.svg
  - Now
    - /images/camera.#008000.16x16.ccsvg
- 
- color      size

# In the background...

- Internal processing
  - „ccsvg“ is bound to servlet „DynamicImageServlet“
  - Class „SVGImageManagerFactory“ creates instance of „SVGImageManager“
    - Own class by registering via „initSVGManagerClass(Class clazz)“
  - Class „SVGImageManager“ updates the SVG according to color and size definition
- Please check your icon library, if the internal processing is „good enough“ to update the SVG accordingly

# And one little issue

- You can use „@<style\_values>@“ inside the .ccsvg image path:
- Original image
  - /images/camera.svg
- „Hard way“
  - /images.camera.#008000.16x16.ccsvg
- „Soft way“
  - /images.camera.@svgDark@.@svgNormalSize@.ccsvg

# CC and the image size (I)

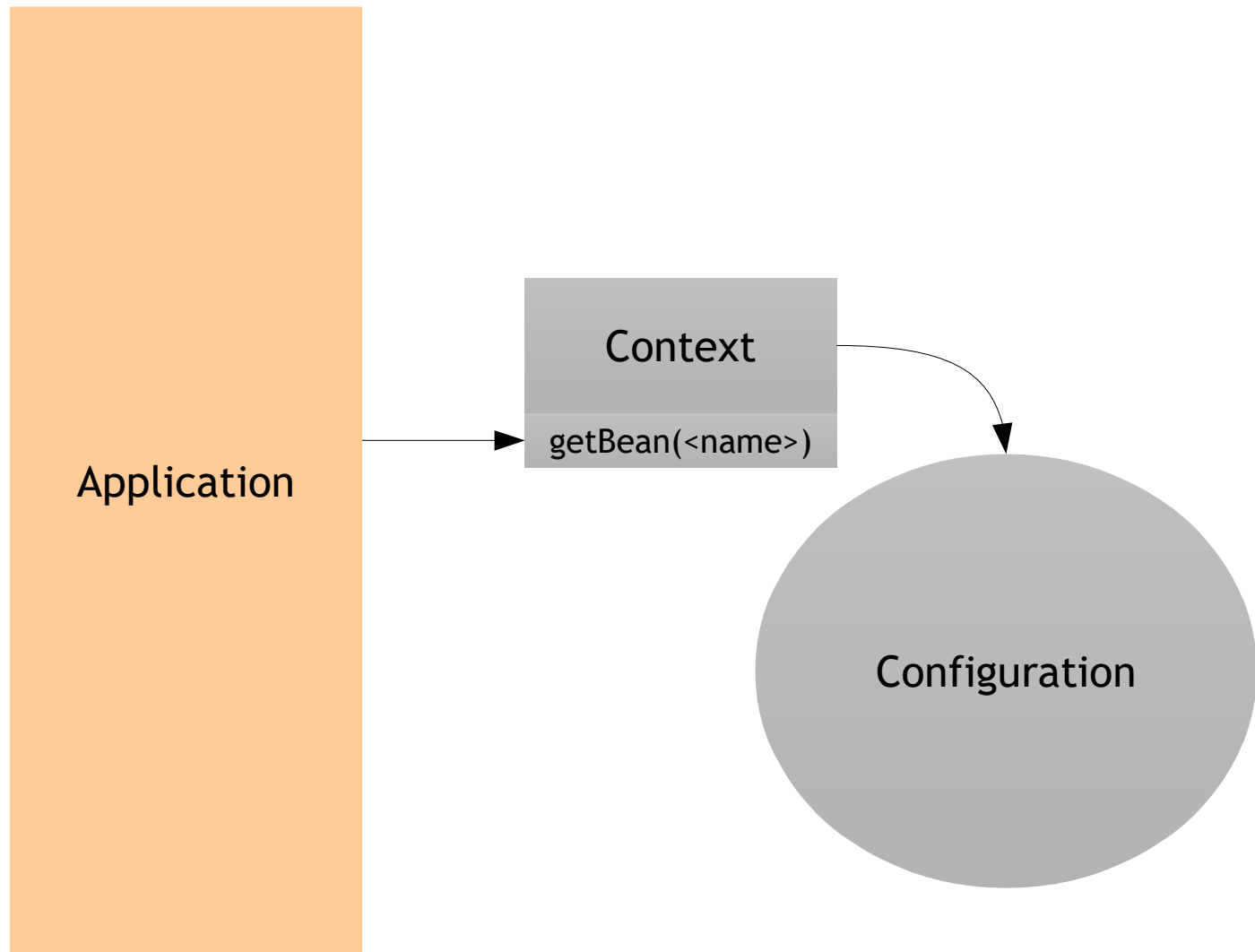
- The client processing requires to know the size of an image
  - ...which is difficult to **synchronously** find out in HTML/JavaScript
- Consequence: if the size of an image is required, the server is requested to find out!
  - Servlet: ImageSizeServlet
  - <http://localhost:50000/cm2019/ccimagesize/images/zzz1.svg>

## CC and the image size (II)

- You may help the client by encoding the default image size into the name of the image
  - /images/camera\_**16x16**.svg

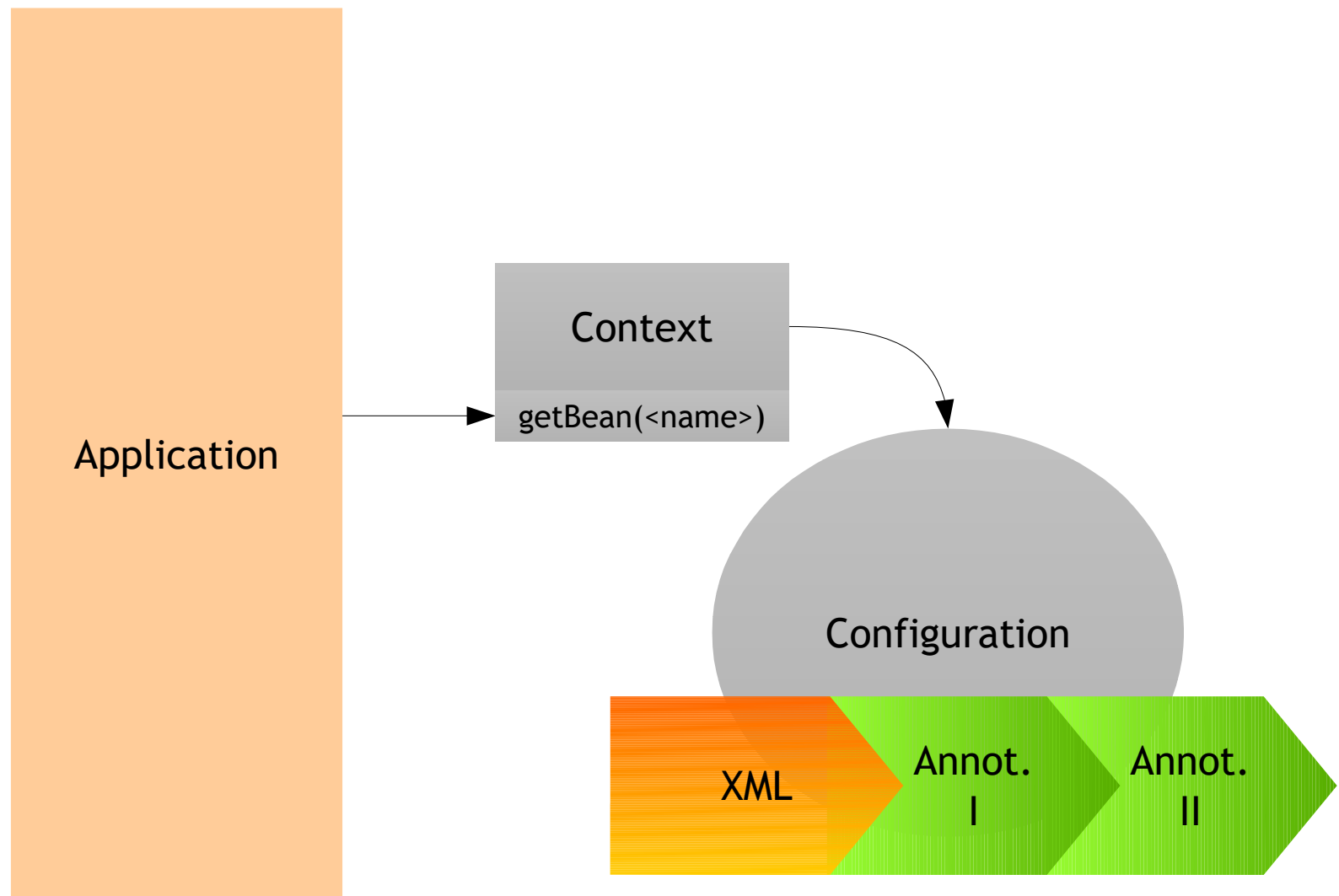
# Spring

# In principal, Spring is easy...



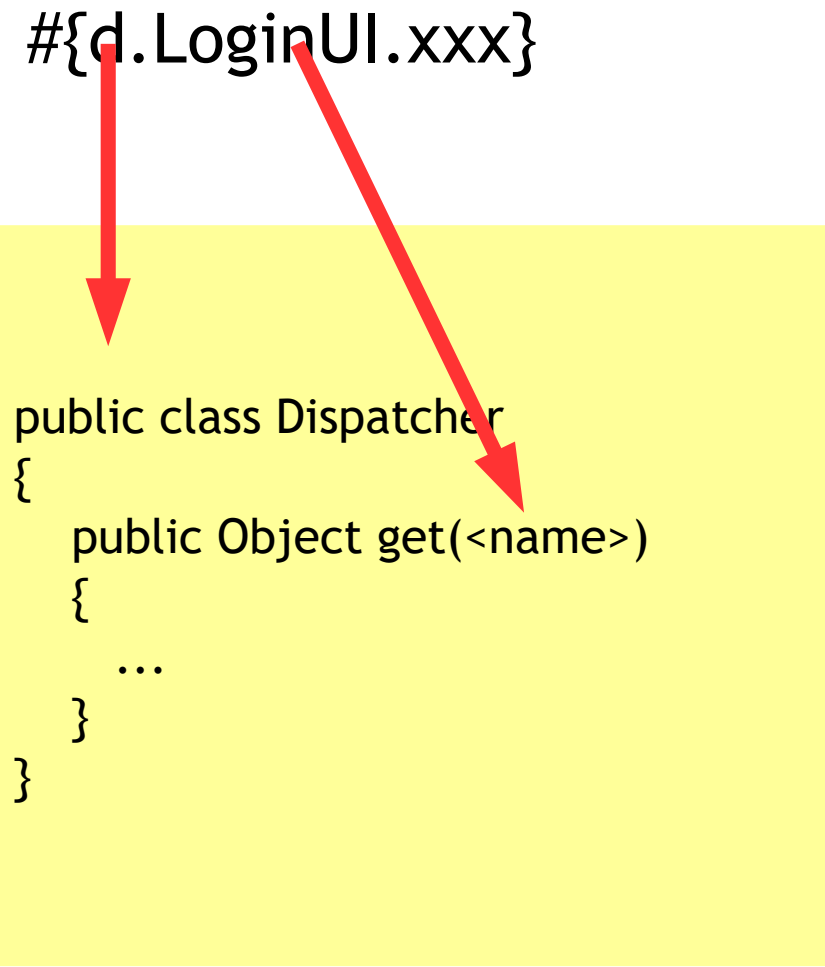


# In principal, Spring is easy...



# Spring inside CaptainCasa I

- `#{d.LoginUI.xxx}`

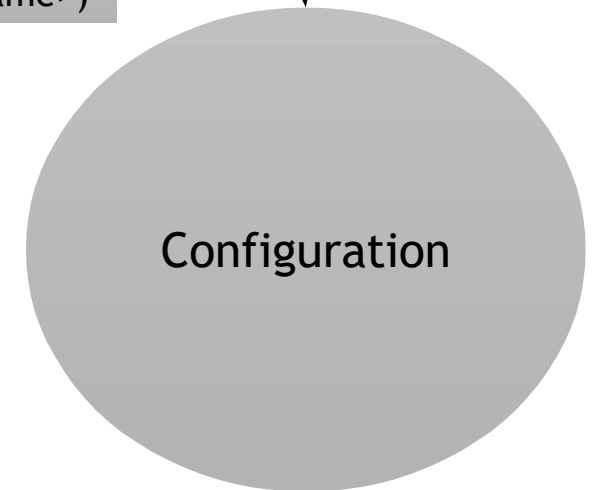
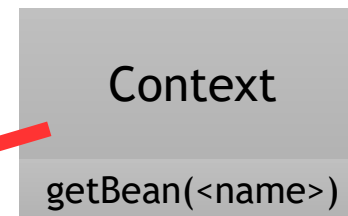


```
public class Dispatcher
{
    public Object get(<name>)
    {
        ...
    }
}
```

# Spring inside CaptainCasa - Idea 1

- `#{d.LoginUI.xxx}`

```
public class Dispatcher
{
    public Object get(<name>)
    {
        ...
    }
}
```



# Spring inside CaptainCasa - Idea 1

- Example: Configuration of PageBeans via Spring
- Advantages
  - Less code
  - More structure
- Disadvantages
  - Debugging
  - Complexity/Readability
- Complexity is moved
  - ...but not removed...!

# Spring inside CaptainCasa I

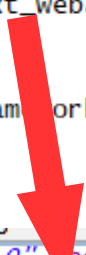


```
.....  
..... public DialogSessionXMLApplicationContext() throws BeansException  
..... {  
.....     super();  
.....     webApplicationContext = WebApplicationContextUtils.getWebApplicationContext(Http  
.....     setParent(wac);  
.....     setClassLoader(HotDeployManager.currentClassLoader());  
.....     CLog.L.log(CLog.LL_INF, "Creating Spring DialogSessionXMLApplicationContext.");  
..... }  
.....  
..... //-----  
..... // public usage  
..... //-----  
.....  
..... public String getConfigLocation() { return m_configLocation; }  
..... public void setConfigLocation(String configLocation) {  
..... {  
.....     m_configLocation = configLocation;  
.....     setConfigLocations(configLocation);  
.....     refresh();  
..... }  
..... }  
.....
```

# Spring inside CaptainCasa - Idea II

web.xml

```
51 .....<context-param>␣␣␣
52 .....<param-name>contextConfigLocation</param-name>␣␣␣
53 .....<param-value>classpath:context_webapplication.xml</param-value>␣␣␣
54 .....</context-param>␣␣␣
55 ␣␣␣
56 .....<listener>␣␣␣
57 .....<listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>␣␣␣
58 .....</listener>␣␣␣
59 ␣␣␣
```



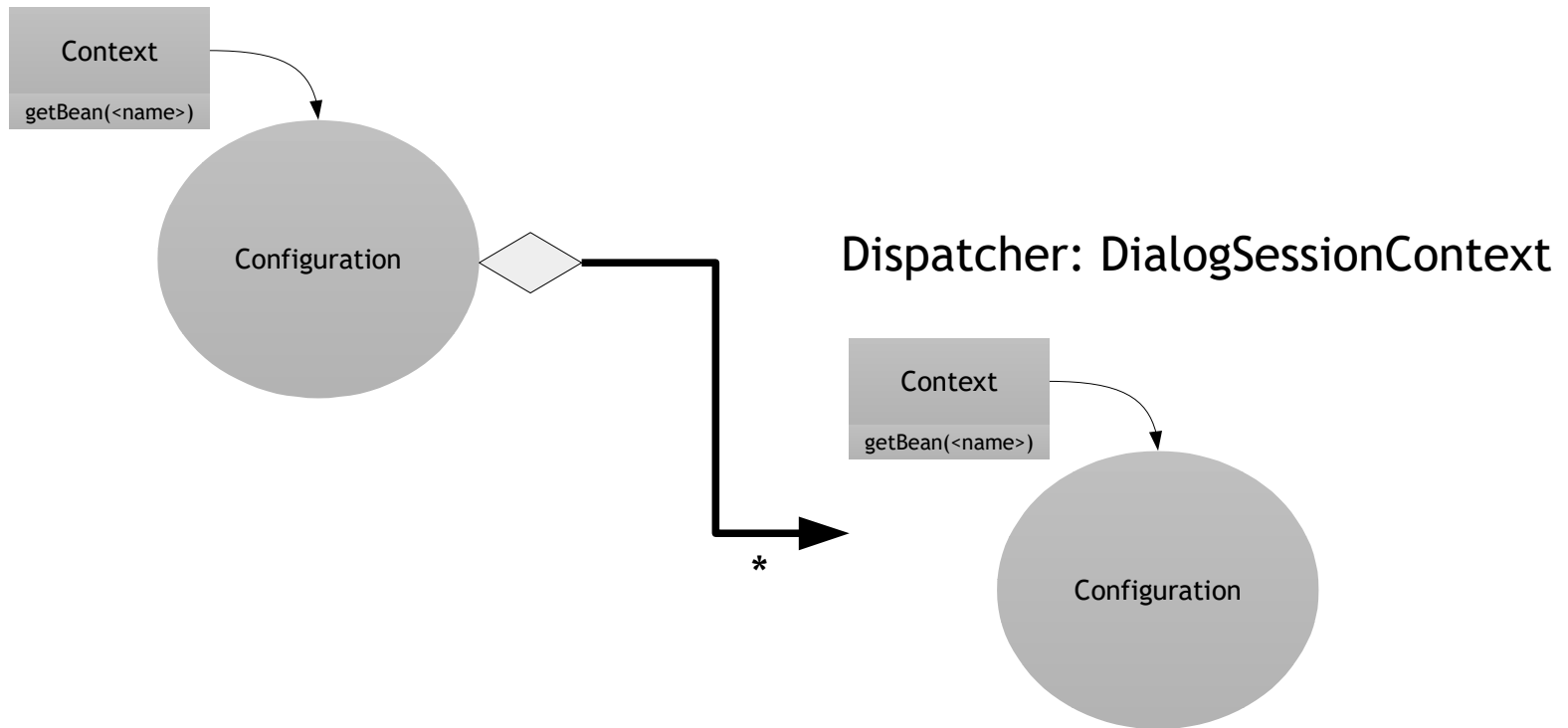
```
1 <?xml version="1.0" encoding="UTF-8"?>␣␣␣
2 <beans xmlns="http://www.springframework.org/schema/beans"␣␣␣
3 .....xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"␣␣␣
4 .....xsi:schemaLocation="http://www.springframework.org/schema/beans␣␣␣
5 .....https://www.springframework.org/schema/beans/spring-beans.xsd">␣␣␣
6 ␣␣␣
7 .....<bean id="DialogSessionApplicationContext"␣␣␣
8 .....class="org.eclnt.spring.context.DialogSessionXMLApplicationContext"␣␣␣
9 .....scope="prototype">␣␣␣
10 .....<property name="configLocation" value="springuibears/context_dialogsession.xml"/>␣␣␣
11 .....</bean>␣␣␣
12 .....␣␣␣
13 </beans>␣␣␣
14 ␣␣␣
```

In Java

```
AbstractApplicationContext c = DialogSessionApplicationContextFactory.instance();
```

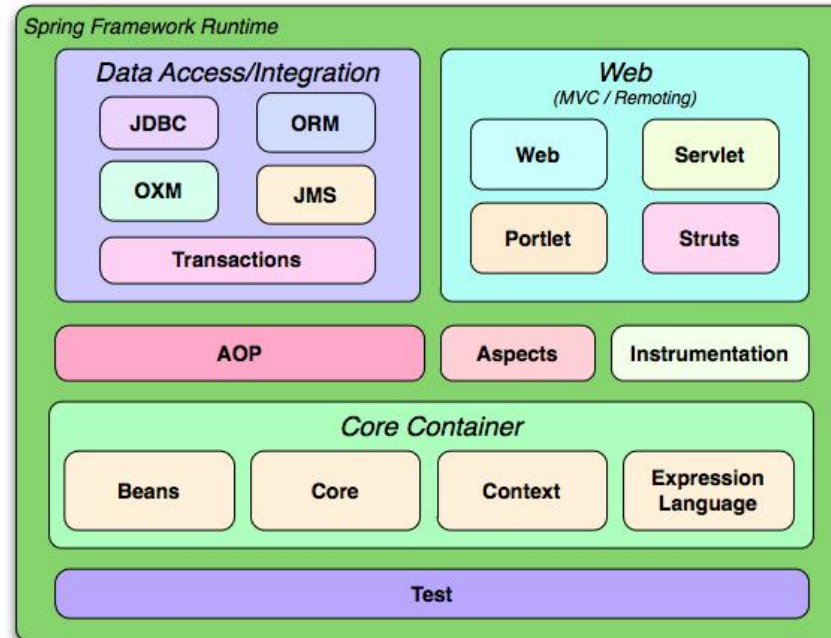
# Idea I combined with Idea II

web.xml/Spring: WebApplicationContext



# This was the core of Spring, but...

- This was the core of Spring, but...
  - Based on this coupling of objects Spring built up a service infrastructure for web/application management





# And now complexity starts...

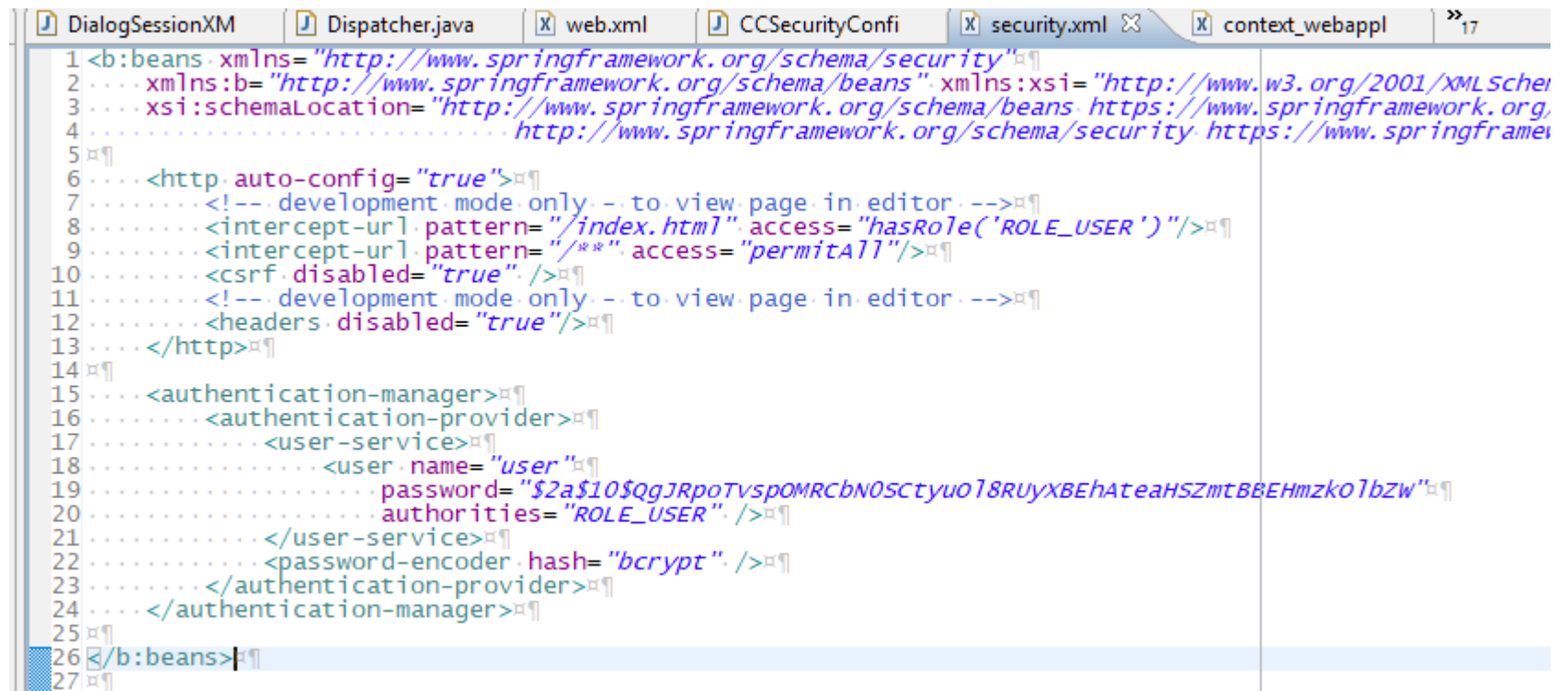
- Each of the services is complex
  - Because it solves some complex problem!
- Documentation is dependent on the service
  - Example: Spring 5 „OpenId Connect“
    - Some too short tutorials
    - StackOverflow as main documentation
  - Configuration: XML, 2 \* Annotations
- Configuration is complex
  - Configuration approach requires deep understanding of object structure
  - May depend on Spring version

# Example: adding Spring security

web.xml

```
55 |  
56 |... <filter>|  
57 |... <filter-name>springSecurityFilterChain</filter-name>|  
58 |... <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>|  
59 |... </filter>|  
60 |... <filter-mapping>|  
61 |... <filter-name>springSecurityFilterChain</filter-name>|  
62 |... <url-pattern>/*</url-pattern>|  
63 |... </filter-mapping>|  
64 |
```

security.xml



```
DialogSessionXM | Dispatcher.java | web.xml | CCSecurityConfi | security.xml | context_webappl | »17  
1 | <b:beans xmlns="http://www.springframework.org/schema/security"|  
2 | ... xmlns:b="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w3.org/2001/XMLSchema  
3 | ... xsi:schemaLocation="http://www.springframework.org/schema/beans https://www.springframework.org,  
4 | ... http://www.springframework.org/schema/security https://www.springframe  
5 | |  
6 | ... <http auto-config="true">|  
7 | ... <!-- development mode only -- to view page in editor -->|  
8 | ... <intercept-url pattern="/index.html" access="hasRole('ROLE_USER')"/>|  
9 | ... <intercept-url pattern="/*" access="permitAll"/>|  
10 | ... <csrf disabled="true"/>|  
11 | ... <!-- development mode only -- to view page in editor -->|  
12 | ... <headers disabled="true"/>|  
13 | ... </http>|  
14 | |  
15 | ... <authentication-manager>|  
16 | ... <authentication-provider>|  
17 | ... <user-service>|  
18 | ... <user name="user"|  
19 | ... password="$2a$10$QgJRpoTVspOMRCbN0Sctyu078RUYXBEhAteaHSZmtBBEHmzko7bZW"|  
20 | ... authorities="ROLE_USER"/>|  
21 | ... </user-service>|  
22 | ... <password-encoder hash="bcrypt"/>|  
23 | ... </authentication-provider>|  
24 | ... </authentication-manager>|  
25 | |  
26 | </b:beans>|  
27 |
```

# My wish...

- Spring provides a lot of infrastructure functions for managing/securing web applications
- Let us use these functions and let us gain/share the knowledge how to use these functions
  - Bulgaria in action!
- Examples
  - Spring SAML implementation
  - Spring OpenId Connect implementation

**CCEE**

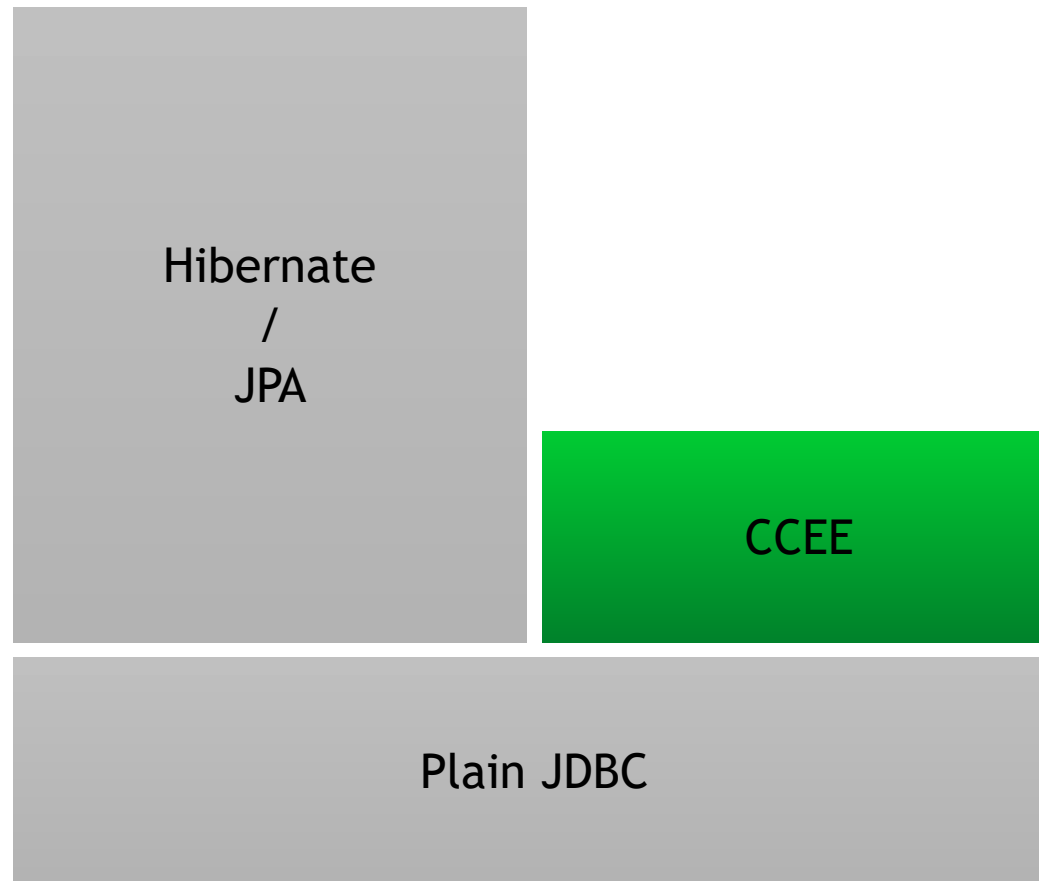
# CCEE = CaptainCasa Enterprise Edition

- Collection of useful frameworks to build a business application
- Areas
  - Configuration
  - **Persistence Management**
  - Scheduling
  - Solr based text search
  - ...and others

# CCEE Persistence Management

- Very thin, very straight forward layer for persisting Pojo-Objects
- Why not Hibernate/JPA?
  - Nice
    - Object mapping
    - DB abstraction
  - Not nice:
    - Combination with Context/Session management
    - Implicit navigation...
  - Hidden Complexity!

# CCEE Persistence Management



# CCEE Persistence Management

- ...focusing on streamlining all SQL operations that deal with one table!
  - Mapping Pojo-Properties <> Table Columns
  - Insert/update/delete
  - Query
- ...possibility to add any free style SQL by directly assembling SQL statements
- NO object factory
- NO implicit navigation



# Supported data types

## Pojo Types

## JDBC Types

int, Integer	Integer
byte, Byte, long, Long	Byte, Long
float, Float, double, Double	Float, Double
LocalDate	java.sql.Date
LocalTime	java.sql.Time
LocalDateTime	java.sql.Timestamp
Date	java.sql.Timestamp
java.sql.Date/Time/Timestamp	java.sql.Data/Time/Timestamp
BigDecimal	BigDecimal
BigInteger	Long
boolean, Boolean	Boolean
UUID	String ("d3c26822-70d8-4a1d-977f-394b04e0fd67")
byte[]	byte[]

# DOFWSql

- query(...)
  - queryColumnData(...)
  - queryTop(...)
  - queryOne(...)
  - queryDistinct(...)
  - queryFunction(...)
- save(...)
  - insert(...)
  - update(...)

# DB „Context name“

- Each context is associated with own database configuration
  - Every DOFWSql function allows to pass the context name
- Purpose: working with multiple databases
  - But: no synchronized commit!

# Job Scheduler

```
package org.eclnt.ccee.quartz.logic;
public interface ICCEEJob
{
    public void executeJob(String parameters, CCEEJobExecutionContext jobExecutionContext);
}
```

The screenshot displays the Quartz Configuration web interface. At the top, there are tabs for "Monitoring", "Configuration", and "Database schema". The "Monitoring" tab is active, showing a start time of "03.06.2019 12:55:30" and a frequency of "TT.MM.JJJJ SS:MM:SS". A button labeled "Ausgeführte Jobs anzeigen" is visible. Below this, a table lists jobs with columns for "Job Id" and "Status". All jobs listed are "OutputAllAttendees" and have a status of "ENDED".

An overlay window titled "Job execution protocol" is open, showing details for a specific job execution. The "Id" is "159f77a9-a0d1-4783-951e-9ed0775ba7a8" and the "Status" is "ENDED". The "Protokoll" section contains several log entries, each starting with "INFO | Found attendee:" followed by a UUID. The entries are:

- INFO | Found attendee: 788381fc-2723-462a-8a0a-a2fa1d2fdac9
- INFO | Found attendee: e7aa44b9-f334-4e36-8fd8-ca998a131feb
- INFO | Found attendee: 83c7d504-bcc8-4ba5-848a-bbcf7361ad50
- INFO | Found attendee: e87eb380-f412-4645-a6d7-e00d67dc6930
- INFO | Found attendee: 4da20a15-3948-4f04-b5f6-7130859e9d57
- INFO | Found attendee: 66918fe5-c8b6-4ebb-83fb-10af88554a8d
- INFO | Found attendee: 9f8891bb-9366-4222-8463-b317bd0e7c2a
- INFO | Found attendee: f3c104ad-0e83-4583-a93e-a29d1bc9e2cb
- INFO | Found attendee: eea878e7-c037-4499-8a6a-

The window also includes an "Abbrechen" button at the bottom right.

# Job Scheduler

- Internally is based on „Quartz“
- Update of Job-execution-tables in same transaction than application

# CloneUtil

```
.....  
... public static <CLASS extends Serializable> CLASS createClone(CLASS o)  
... {  
..... try  
..... {
```

# Containers and Sizing

# Default Container (PANE, ...)

- „Pixel“-sizing
  - width="100"
  - width="100+"
- Percentage-sizing
  - width="50%"
  - width="50%;200"
  - cutwidth="true"
- (same with height)



# Layering

- OVERLAYAREA
  - OVERLAYAREAITEM
    - x,y,width,height
    - shownonareamousover

# Filling lines...

- ROWFLEXLINECONTAINER
  - Columns: ROWFLEXCOLUMNCONTAINER
- ROWMATRIXAREA
- ROWADAPTIVEAREA

# Animations

# CAPTUREANIMATOR

- PANE

CAPTURANIMATOR

trigger

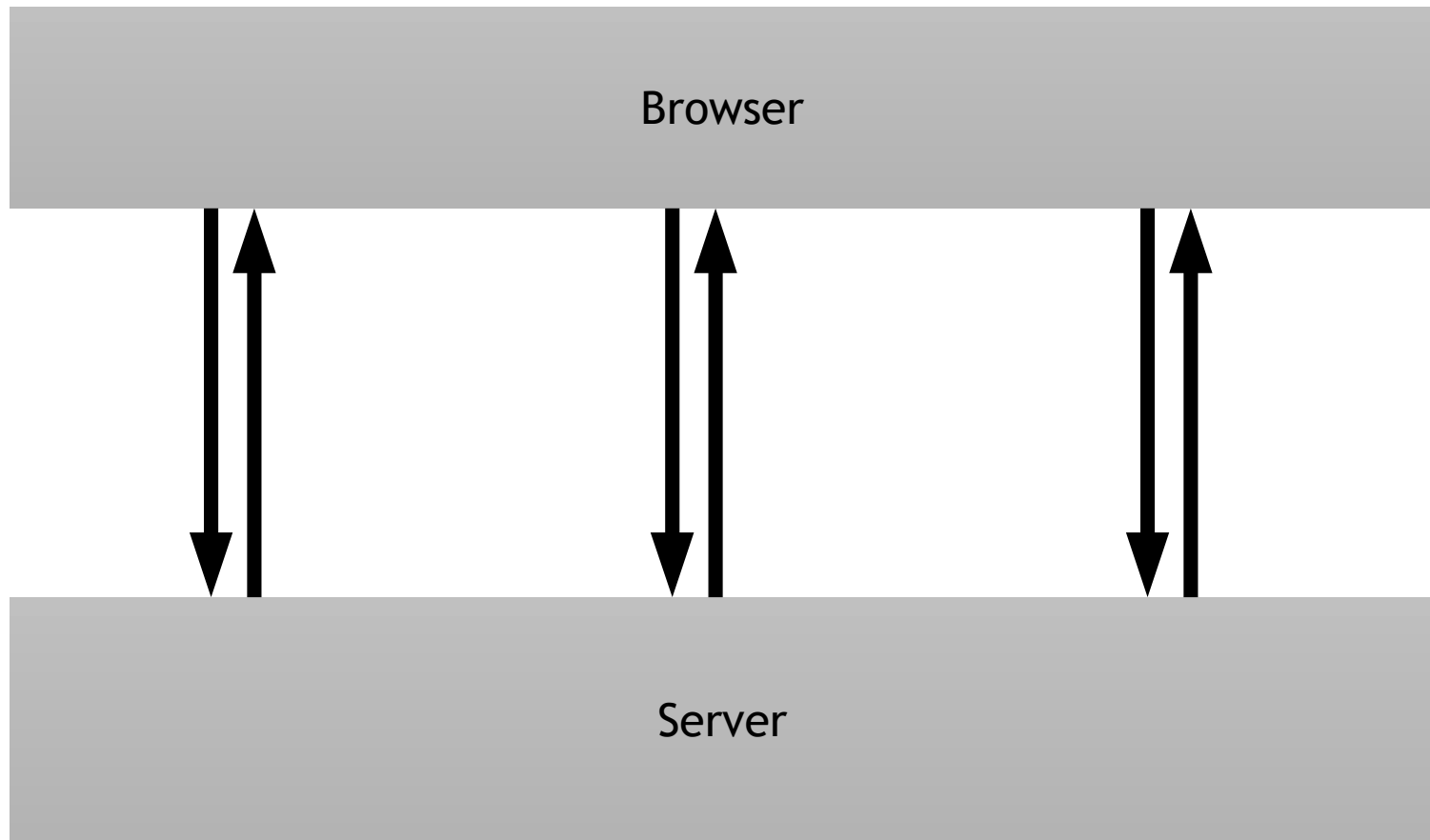
animationtype

# ANIMATECHANGEOSIZE

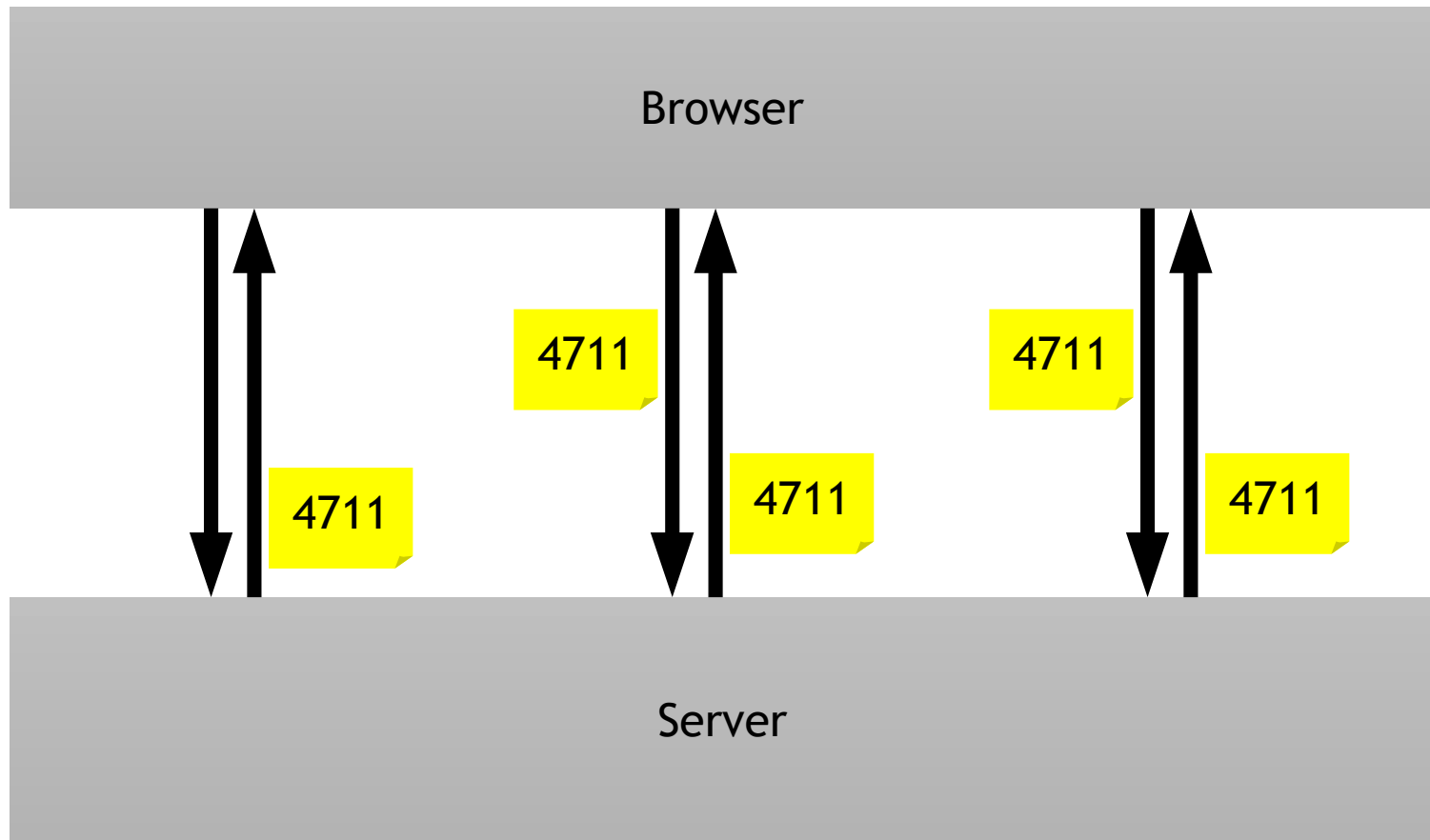
- ...with any component!

# Session Management

# The principal problem



# The principal problem





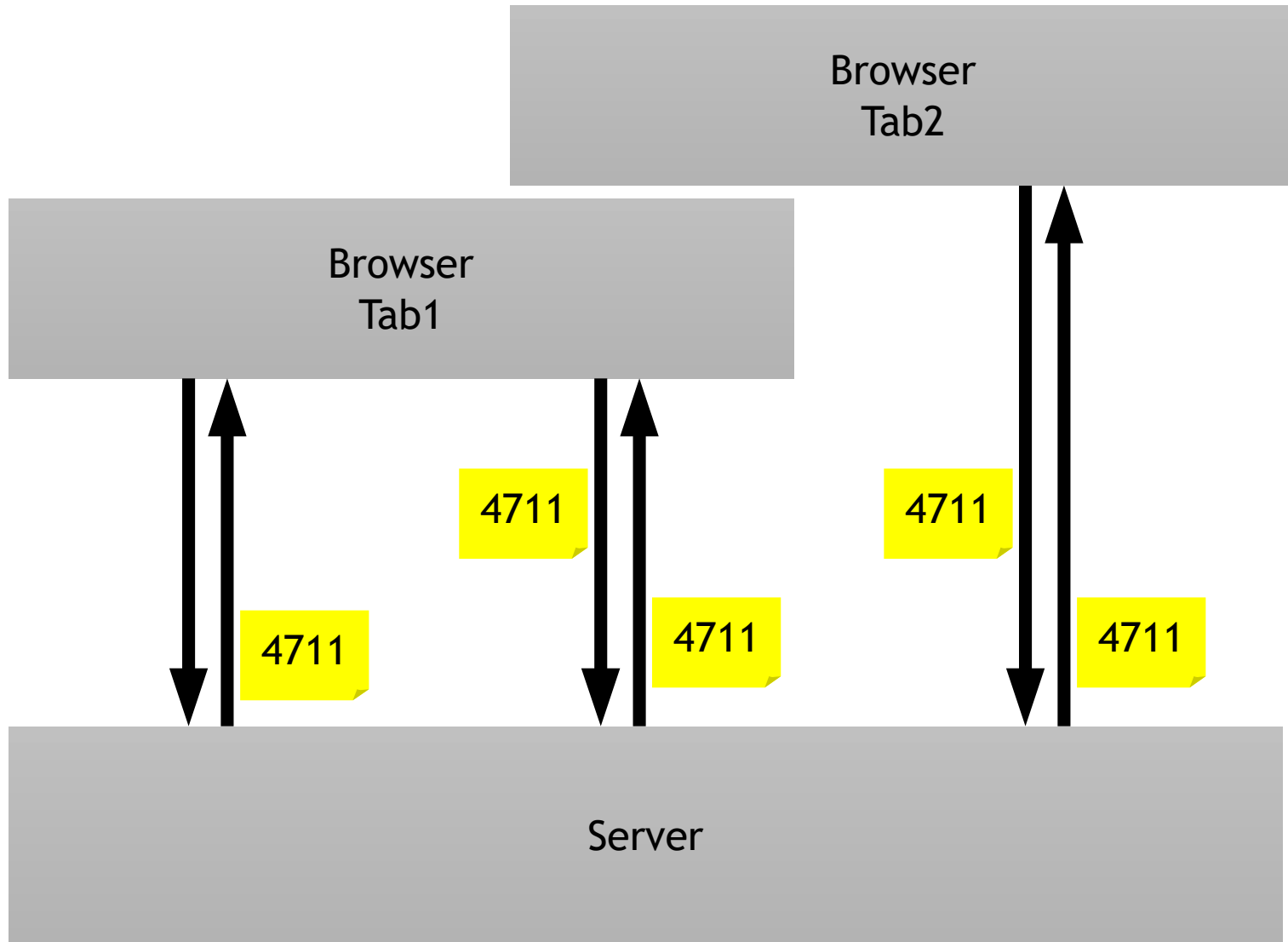
# Default: SessionId encode into URLs, no Cookies

- Any URL that is passed from the server to the client in order to continue the session incorporates some encoded sessionId
  - `http://...../whatever/xyz.html;jsessionId=4711`
- Advantage
  - No Cookie required!
  - No problem to have individual http-sessions per browser tab
- Disadvantage
  - No Cookie required!
  - Problems with almost all SSO frameworks...

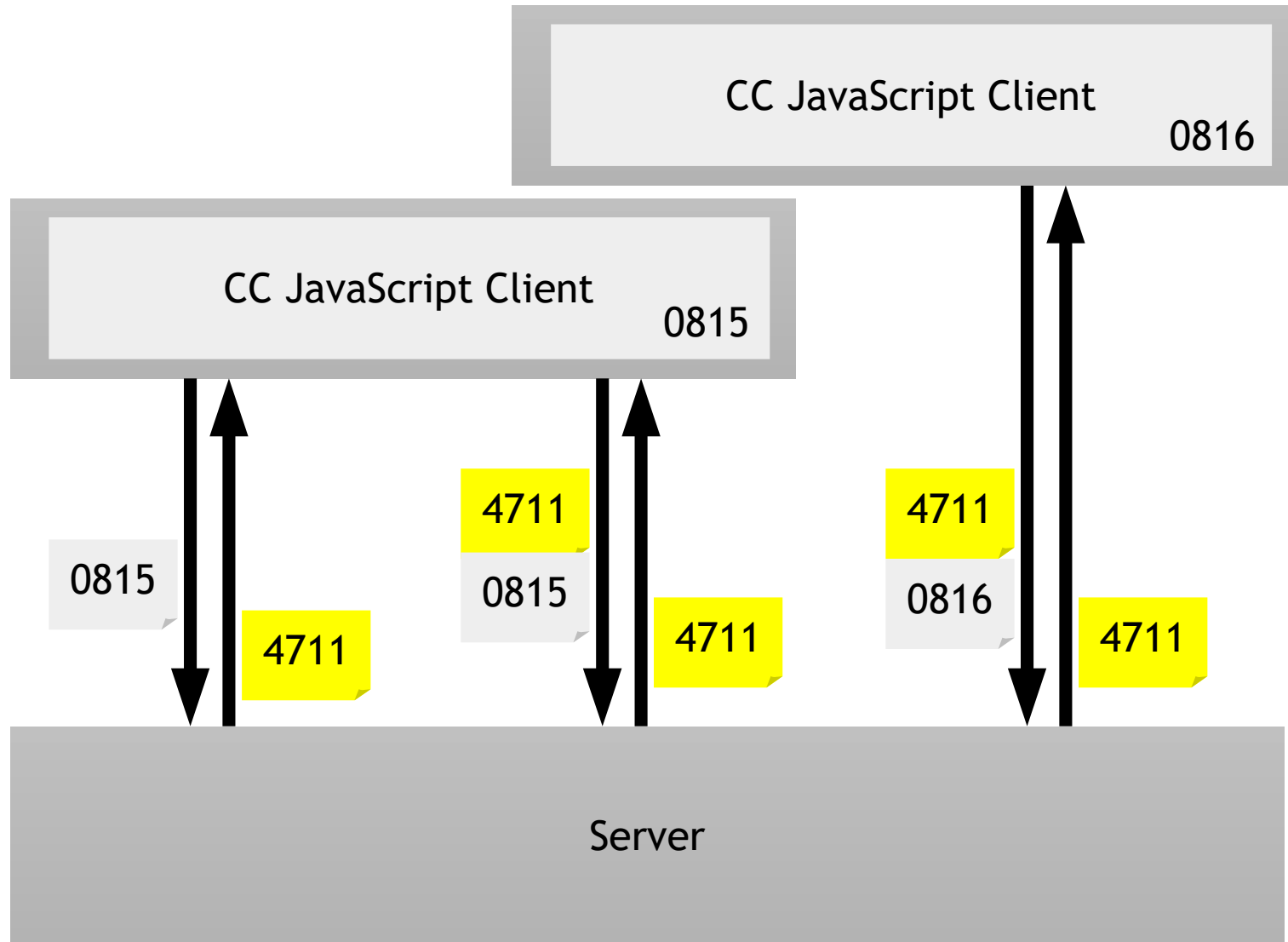
# Cookie Problem

- The „normal default“ today is session management by Cookie
- But: Cookies are shared across browser tabs!
  - Cookies are nice to transfer authentication session information - but are not usable to identify one browser tab!!!

# Cookie Problem



# Luckily, CaptainCasa keeps a „Subpage-Id“



# What the server does with the SubpageId...

- Any expression is (and was always!) resolved by a CC-Expression-Resolver
  - #{d.WhateverUI.firstName}

```
.....@Override
.....public Object getValue(ELContext context,
.....                        Object base,
.....                        Object property)
.....    throws NullPointerException, PropertyNotFoundException, ELException
.....    {
.....        // "Variable Resolver Processing"
.....        if (base == null && property != null)
.....        {
.....            if (m_firstResolving == true)
```

- This is the place where the lookup of e.g. „d“ is done per SubpageId

# Consequence

- Switching to Cookie-based session management is possible
  - Objects of one tab (subpageId) are clearly separated from objects of another tab

web.xml

```
344 ..  
345 ..<!-- ***** SESSION MANAGEMENT *****  
346 ..  
347 ..<session-config>  
348 .....<session-timeout>60</session-timeout>  
349 .....<tracking-mode>COOKIE</tracking-mode>  
350 ..</session-config>  
351 ..
```

system.xml

```
13 .....  
14 .....-->  
15 ....<sessionmanagement  
16 .....type="COOKIE"  
17 ....</>  
18 ..
```

# Two types of sessions

- „Classic“      HttpSession      (cross Browser)
- „New“      SubpageSession      (per Browser)
  
- You now have to clearly separate in order to be able to transfer your application from „URL-mode“ to „COOKIE“-mode
  - In URL-mode      1 HttpSession => 1 subpageSession
  - In COOKIE-mode      1 HttpSession => n subpageSessions
  
- HttpSessionAccess.getDialogSession()
  - always returns back the „per browser session“

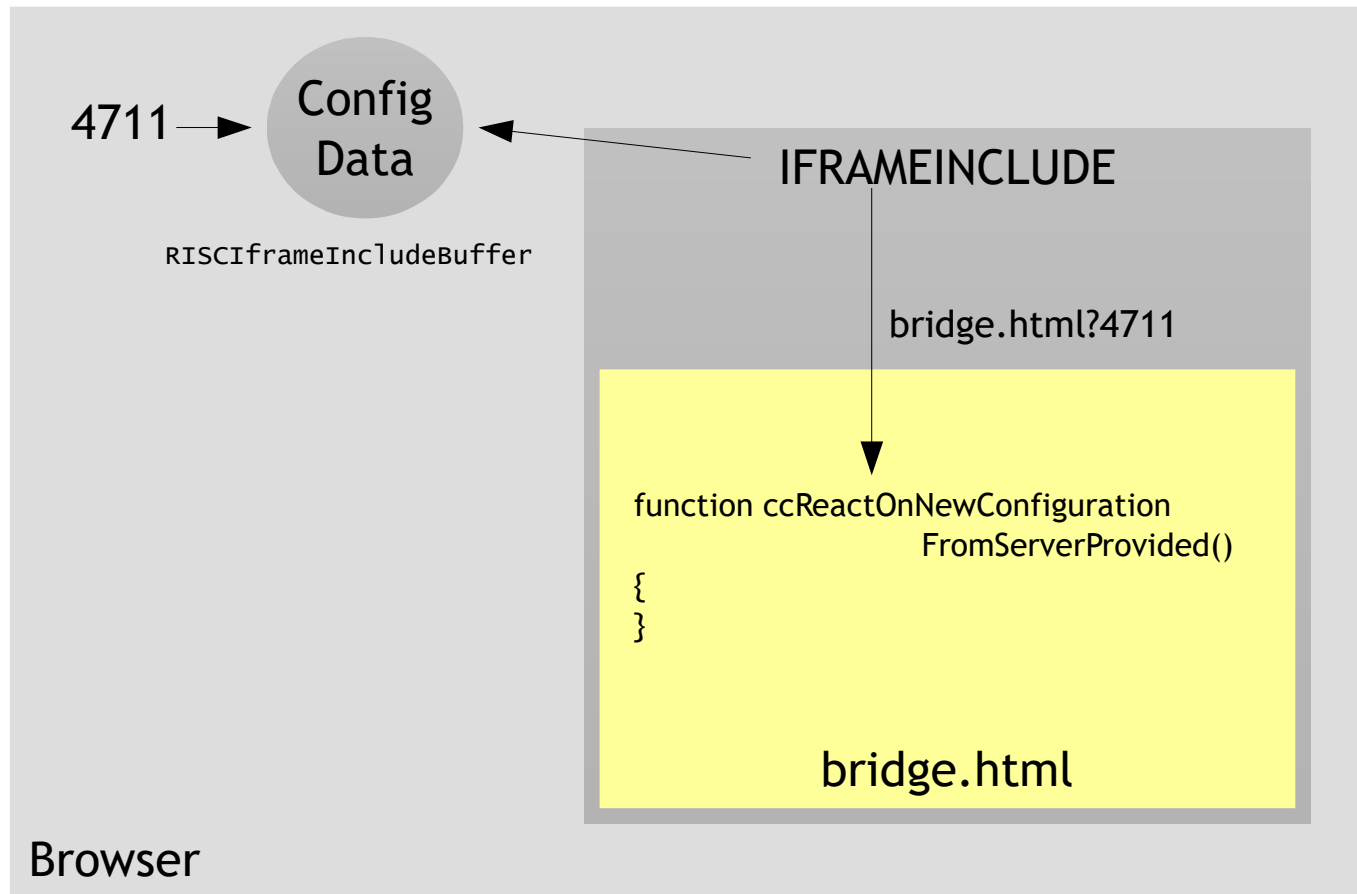
# Integrating other Web Frameworks



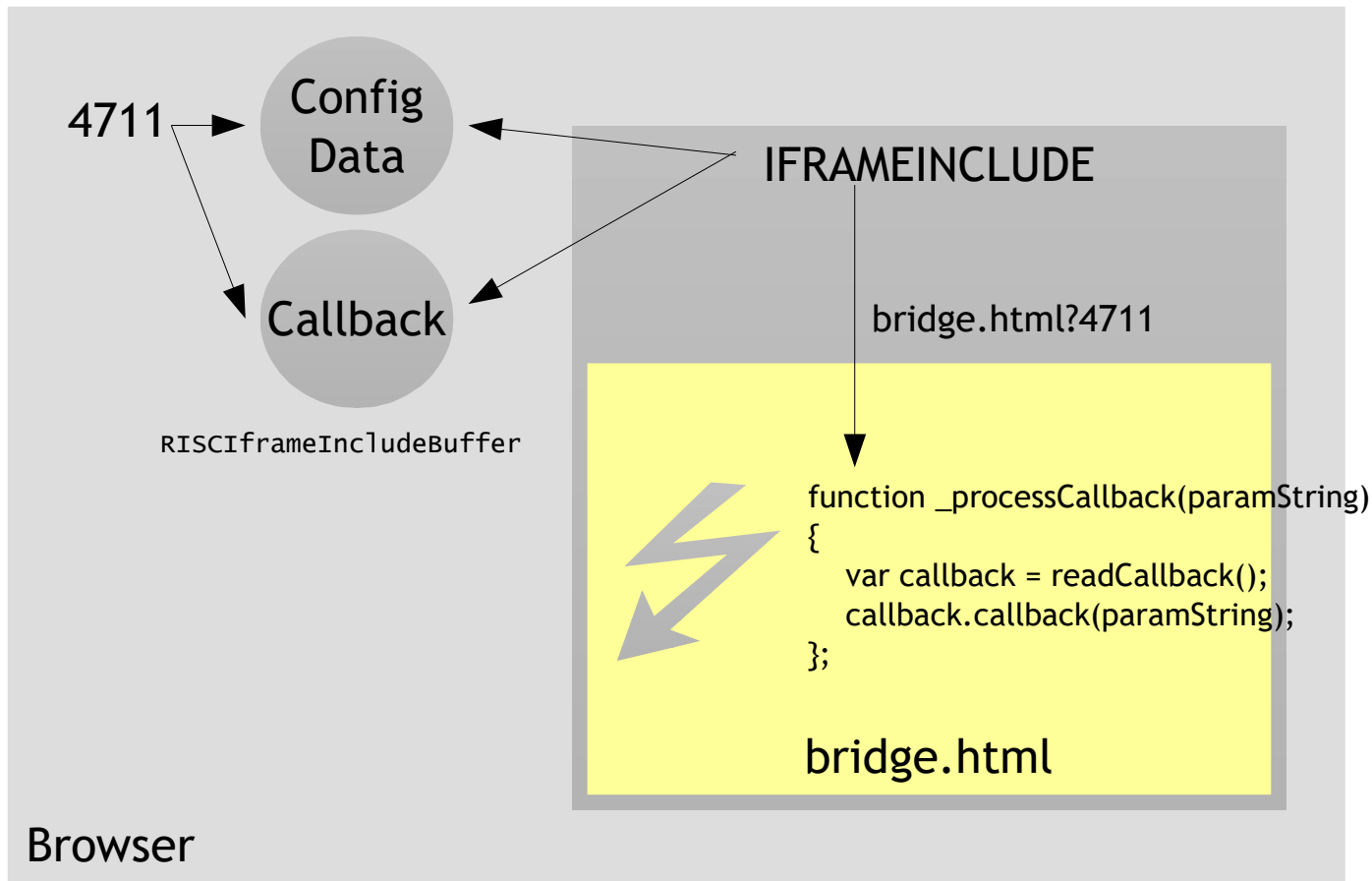
# IFRAME based integration

- Advantages
  - Maximum level of isolation
    - JS libraries
    - CSS
- Disadvantages / Challenges
  - „Heavy approach“
  - Loosing the context when closing/reopening

# IFRAMEINCLUDE component



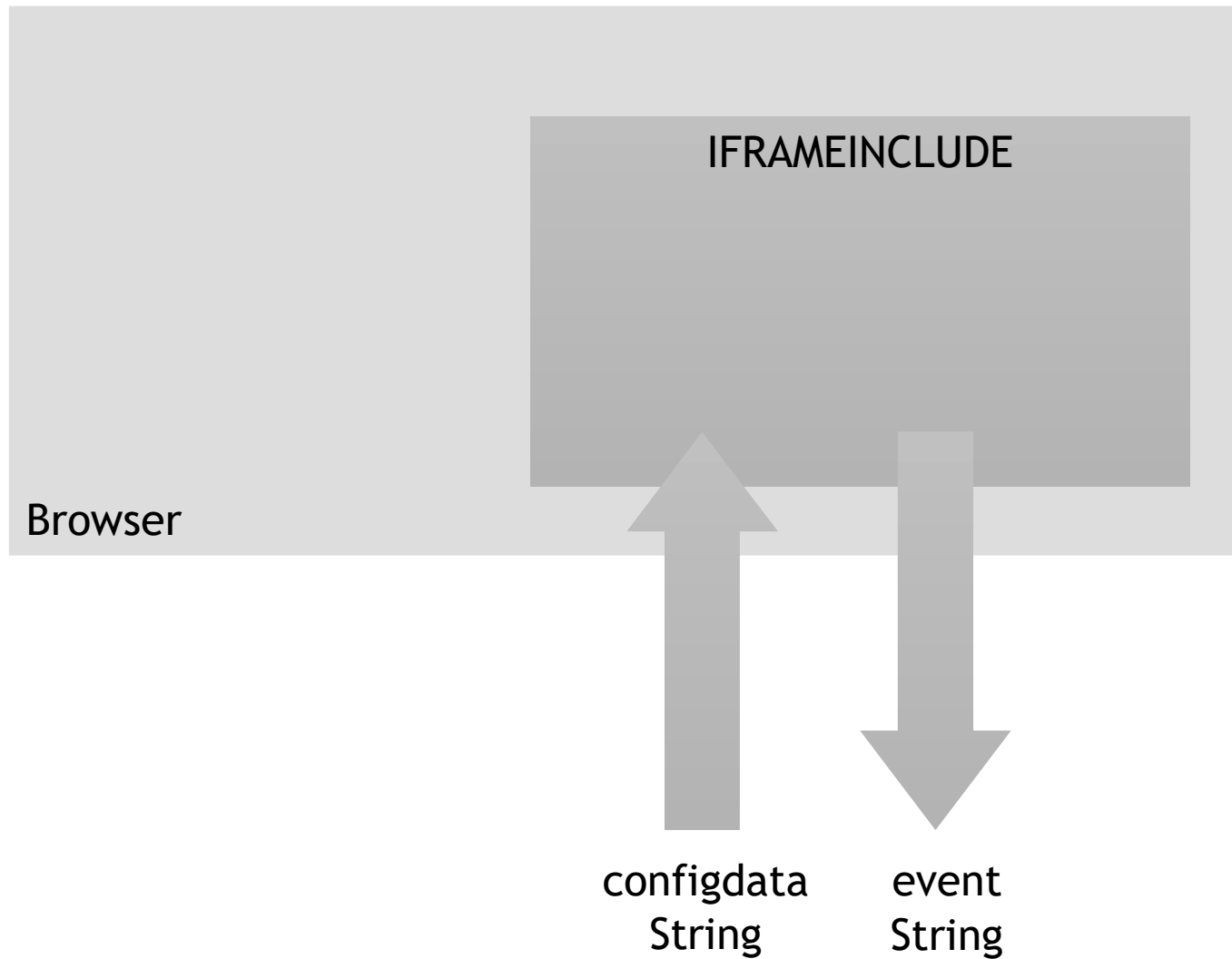
# IFRAMEINCLUDE component



# Typical issues

- Cross site scripting problems?
  - No! - The bridge page is running on the same site than the RISC-page!
- Interim data to be stored in bridge page?
  - Use `RISCIframeIncludeBuffer[„OWNDATA_“+bufferId]`
- Drag/Drop possible?
  - Yes.
- Usable for fine grained controls?
  - No...
  - In this case: write own CC component which is directly part of the RISC page

# Result



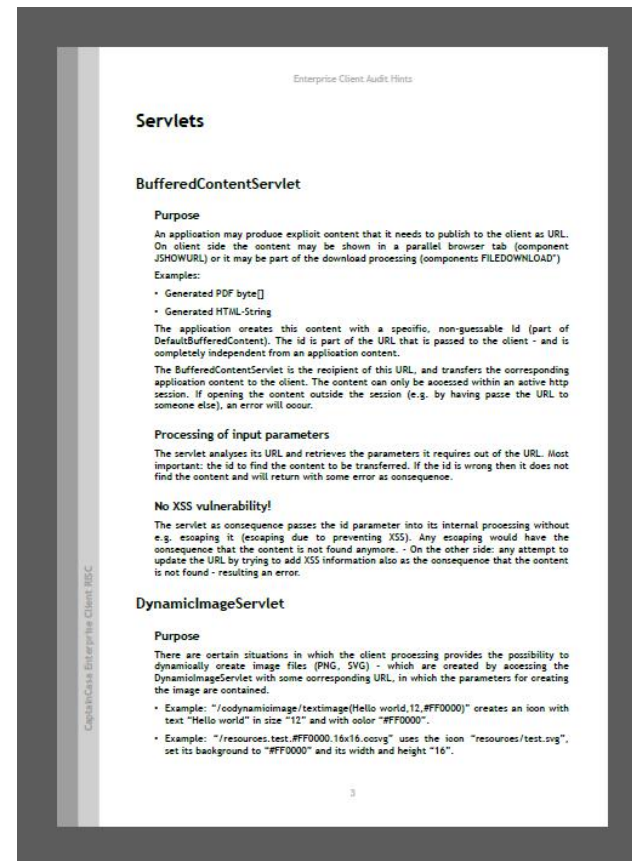
# Audit Survival Guide

# Audit Survival Guide I

- Please...
  - Integrate us into your activities!
    - We are interested, of course!
    - We gained some experience in the meantime
    - ...but please also reserve some budget for integrating us... ;-)

# Audit Survival Guide II

- We documented some of our experiences here:
  - [http://www.captaincasa.com/pdf/eclnt\\_risc\\_audithints.pdf](http://www.captaincasa.com/pdf/eclnt_risc_audithints.pdf)





# Audit Survival Guide III

- Veracode Scans are „industry standard“
  - Static byte code analysis
  - The results are stupid - ...sometimes!
    - Example: Log-Infection
    - ...and Veracode only accepts limited libraries that is trust („ESAPI“)
  - Our lessons learned
    - Follow the VW paradigm: and update the code so that Veracode is satisfied... if Veracode is too weird
    - „Mitigate“ the spots

# Example

Very High	0
High	0
Medium	478
Authorization Issues	
CRLF Injection	356



Flaw Id	Module #	Class #	Module	Location	Fix By
610	6			.../BufferedContentServlet.java 82	
564	6			.../BufferedContentServlet.java 82	
499	6			.../BufferedContentServlet.java 159	
705	6			.../BufferedContentServlet.java 159	
631	6			.../BufferedContentServlet.java 165	
588	6			.../BufferedContentServlet.java 165	
389	9			org/.../util/CCELResolver.java 247	
770	9			org/.../util/CCELResolver.java 247	
455	9			org/.../util/CCELResolver.java 247	
181	9			org/.../util/CCELResolver.java 276	
71	9			org/.../util/CCELResolver.java 276	
210	9			org/.../util/CCELResolver.java 278	
463	9			org/.../util/CCELResolver.java 278	
327	9			org/.../util/CCELResolver.java 316	

Clog.L.log(INF, "...." + xyz)

# „Solution“

Very High

High

Medium

Authorization Issues

CRLF Injection

```
public class CLogLogger extends Logger {
    ... Logger m_redirectLogger = null;
    ... public CLogLogger(String name, String resourceName) {
    ... super(name, resourceName);
    ... public CLogLogger(Logger redirectLogger) {
    ... super(redirectLogger.getName()+"_secure", redirectLogger.
    ... m_redirectLogger = redirectLogger;
    ... @Override
    ... public void log(Level level, String msg) {
    ... if (!isLoggable(level, msg)) return;
    ... update msg if required
    ... String clean = ESAPI.encoder().encodeForHTML(msg);
    ... if (!msg.equals(clean)) clean += " (cleaned)";
    ... if (m_redirectLogger == null)
    ... super.log(level, clean);
    ... else
    ... m_redirectLogger.log(level, clean);
    ... }
```

Fix By

Clog.L.log(INF,“....“ + xyz)

770	9	org/.../util/CCELResolver.java 247
455	9	org/.../util/CCELResolver.java 247
181	9	org/.../util/CCELResolver.java 276
71	9	org/.../util/CCELResolver.java 276
210	9	org/.../util/CCELResolver.java 278
463	9	org/.../util/CCELResolver.java 278
327	9	org/.../util/CCELResolver.java 316

# Example - XSS Flaw

`http://....//images.camera.#FF0000.16x16.ccsvg`

## DynamicImageServlet

- Take parameters from URL
  - `name`
  - `color`
  - `size`
- Call function:
  - `String svg = createSVGImage(name,color,size);`
- Return svg as response
  - `response.getWriter.println(svg)`

# Example - XSS Flaw

`http://....//images.camera.#FF0000.16x16.ccsvg`

`http://....//<script>alert('Hallo')</script>.camera.#FF0000.16x16.ccsvg`

## DynamicImageServlet

- Take parameters from URL
  - `name`
  - `color`
  - `size`
- Call function:
  - `String svg = createSVGImage(name,color,size);`
- Return svg as response
  - `response.getWriter.println(svg)`



# „Solution“

`http://.... /images.camera.#FF0000.16x16.ccsvg`

`http://.... /<script>alert('Hallo')</script>.camera.#FF0000.16x16.ccsvg`

DynamicImageSer

- Take parameter

- `name`
- `color`
- `size`

- Call function

- `String svg = createSVGImage(name,color,size);`

- Return svg as response

- `response.getWriter.println(svg)`

Add „mitigation comment“ into Veracode scan tool, in which you describe, how the functions ensure that no cross site scripting can occur.

Use ESAPI encoding methods wherever appropriate.  
Here: color, size

# Severe flaws from audit activity

## Examples:

- **DynamicImageServlet**
  - `/dynamicimage/textimage>Hello,1500,#FF0000`
    - now checked for reasonable parameters
  - `/dynamicimage/WEB-INF/web.xml`
    - now only accesses images
- **ClassLoaderResourceReader**
  - `/com.harry.HarryUI.clresource`
    - now only accesses images

# Consequences

- Keep up to date with CaptainCasa releases
- Proposal
  - One selected person of your company to be registered for „Security-Issues-Mailing-List“
    - Communication of flaws via this mailing list

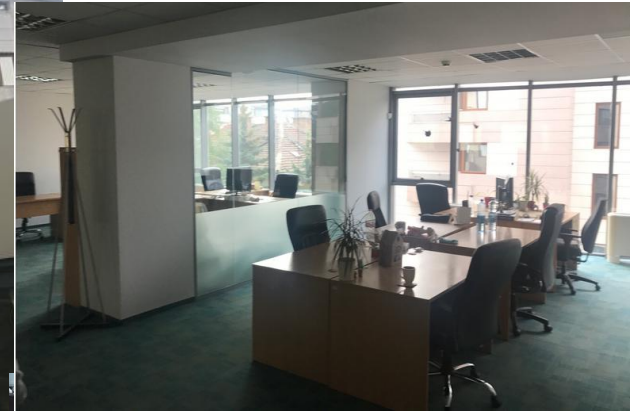


# CaptainCasa Services in Bulgaria

# Evolution IT - The partner in BG



# Sofia





# Team



# Typical project

- ...get to know!
  - Björn, Branimir
  - First PoC by small project 4-6 week
- Kickoff in Sofia
- Close cooperation
  - Two weekly meetings
    - Backlog meeting
    - Status meeting
  - Skyyyyype!
- Quarterly/Half a year meetings in Sofia

# What „Meeting in Sofia means“

- Flight
  - Duration: 2h
  - Cost „WizzAir Priority“: 150 EUR



- Hotel (5 minutes from office)
  - 60 EUR per night



# What „Meeting in Sofia means“

- Restaurants
  - Really cheap (and good!)



- Efficiency
  - 2 ½ days of travel => 1 ½ days of work

# Other issues...

- Rates
  - 390 EUR per day
- Onboarding time
  - 2-4 weeks





# Roadmap 2018

# Roadmap 2019

RISC Client	<b>Components</b> Mobile gestures (sizing) Feature Feature Feature	<b>Install + Project Setup</b> MacOS Linux  Gradle Idea	<b>Infrastructure</b> Embed useful Spring functions in front of CC-processing
Swing	(Maintenance)		
FX	(Maintenance)		
PageBean Components	<b>PageBean Components</b>	<b>CCEE</b>	
Services	<b>Project</b> <b>Project</b> <b>Project</b>		

~~Konfuzius~~ Björn says:

Ctrl-Shift-0

macht froh!