

Vaadin ein Erfahrungsbericht

JUG SWITZERLAND

5. Februar 2013

Benjamin Schupp / Markus Kinzler
mp technology consulting GmbH

Speakers


Benjamin Schupp

Lead Architect, konzipiert und entwickelt Java-basierte Lösungen seit 15 Jahren

Markus Kinzler

Senior Software Engineer, entwickelt seit 10 Jahren mit Java und Web Technologien

mp technology, Zürich – www.mptechology.ch

- Individual-Software für Intranet und Internet und Mobile seit 2003
- Beratung, Analyse, Konzepte, Architektur, Implementierung
- Zertifizierter Vaadin Solution Partner  Solution Partner

Aktuell: Offizielle Vaadin Schulungen in Zürich

- 2. – 3. April: Vaadin Fundamentals
- 4.-5. April: Advanced Vaadin
- <https://vaadin.com/services#vaadintraining>

Agenda

- Motivation
- Vaadin - ein Überblick
- Projektvorstellung
- Einige Anwendungsfälle
- Erkenntnisse und Ausblick
- Fragen / Diskussion

Mission

Umsetzung einer Intranet Geschäftsanwendung zur Planung von Verkaufsfilialen (Einzelhandel)

Rahmenbedingungen:

- Multi-user Web Frontend mit hohem Bedienkomfort, der vergleichbar ist mit herkömmlichen Desktop-Applikationen (RIA)
- MS-Excel gewohnte Nutzer
- Flexible Darstellung grosser Zahlentabellen zum visuellen Vergleich
- Keine graphischen Design Vorgaben
- Moderate Anforderungen an Leistung: < 50 parallele Sitzungen
- Implementierung von Algorithmen zur automatisierten Flächenplanung
- Integration verschiedener IT Systeme der Firma

Anforderungen an das Web-Framework

Interaktive Web-Applikation (RIA)

Umfangreiche Komponentenbibliothek

Schnelle Entwicklung

Werkzeuge / IDE Unterstützung

Wartbarkeit / Erweiterbarkeit / Testbarkeit

Warum Vaadin?



- Organisation
 - Lizenz
 - Firma / Support
- Architektur
 - Server/Client UI
 - Komponentenbibliothek
 - Skalierbarkeit
- Entwicklung
 - Verwendete Programmiersprachen / Programmiermodell
 - Lernkurve / Schnelligkeit
 - Ausbau / Eigenbau von Komponenten
 - Community
 - Dokumentation

Pro Vaadin

Ansprechendes Programmiermodel:

- 100% Java
 - Strenge Typisierung
 - Refactoring
 - Compile-time checking
 - Code completion
 - Automated API docs
 - Kein Javascript / Html Code
 - Browser unabhängig
 - Gute Testbarkeit
- Vereint Vorteile von
 - Server-seitiger Applikationsentwicklung
 - GWT Komponentenbibliothek für Browser

Mit Vaadin

- Versprechungen / Erwartungen

Reifes Framework mit aktiver Community und Doku

Solide Komponentenbibliothek und Add-ons

Einfache und schnelle Entwicklung!

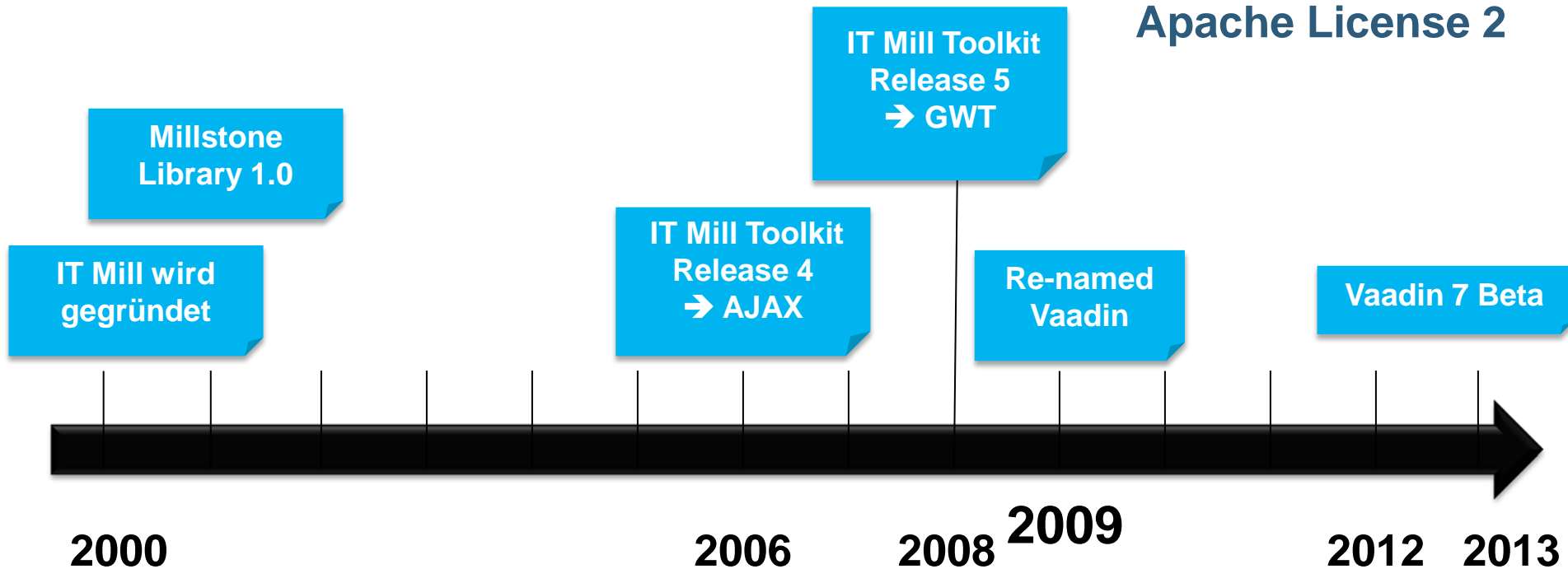
- Kritische Punkte

- Skalierbarkeit?

- Verhalten bei schlechtem Netzwerk?

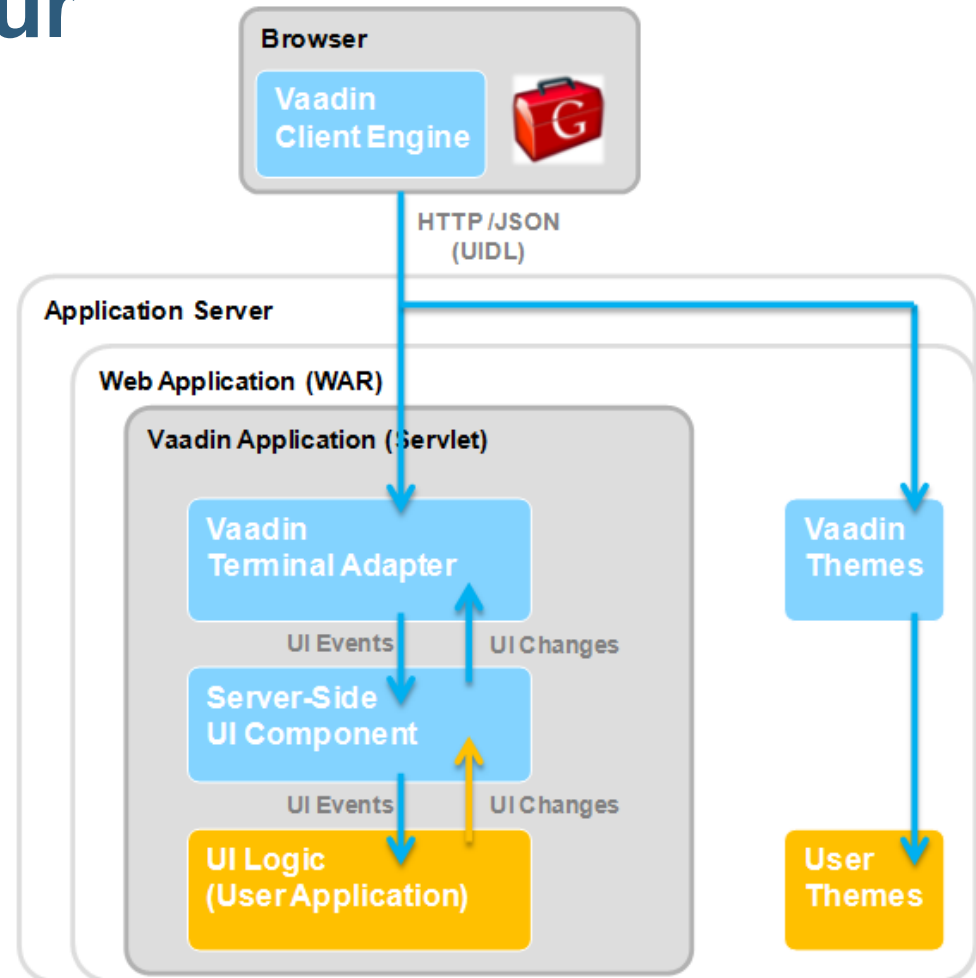
- Komplexität von Entwicklung eigener Komponenten?

Vaadin - Geschichte

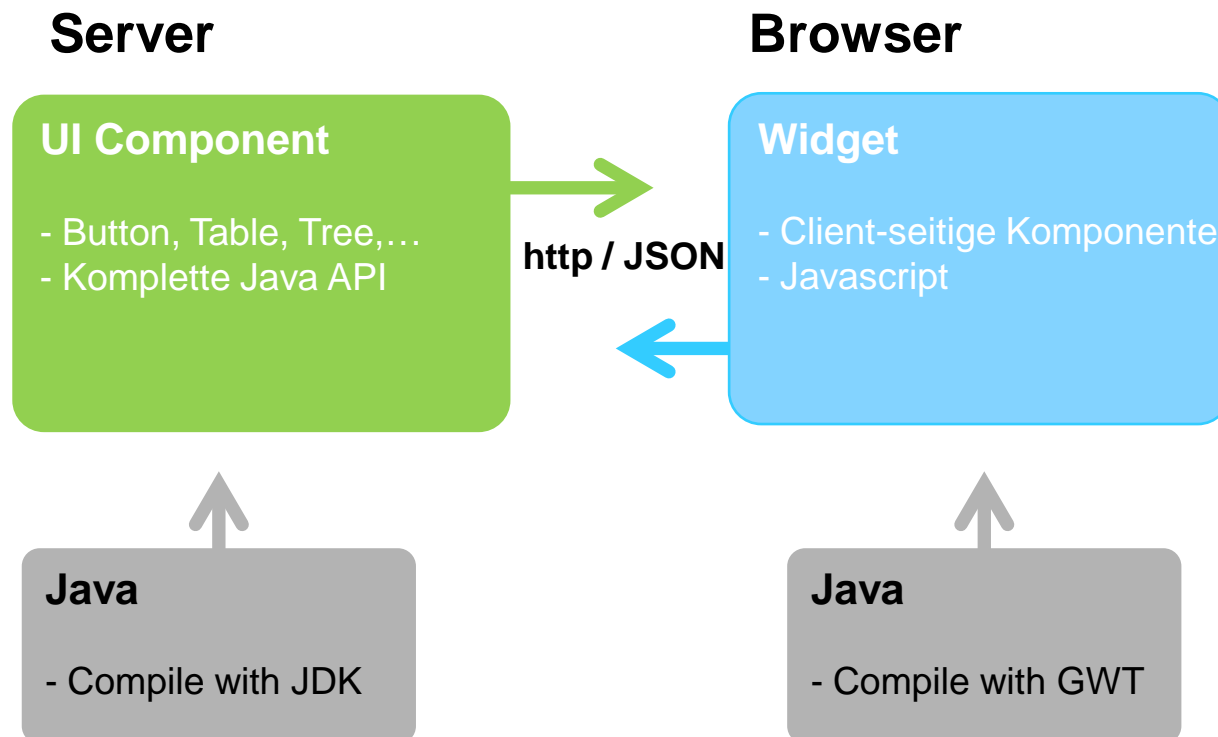


Vaadin - Architektur

- UI Model als Komponentenbaum auf Server
- Rendering im Browser mittels GWT Widgets
- Kommunikation mittels json (UIDL)



Vaadin UI Komponenten Architektur



```
public class SampleApplication extends Application {

    TextField textField;
    Button button;
    int counter = 0;

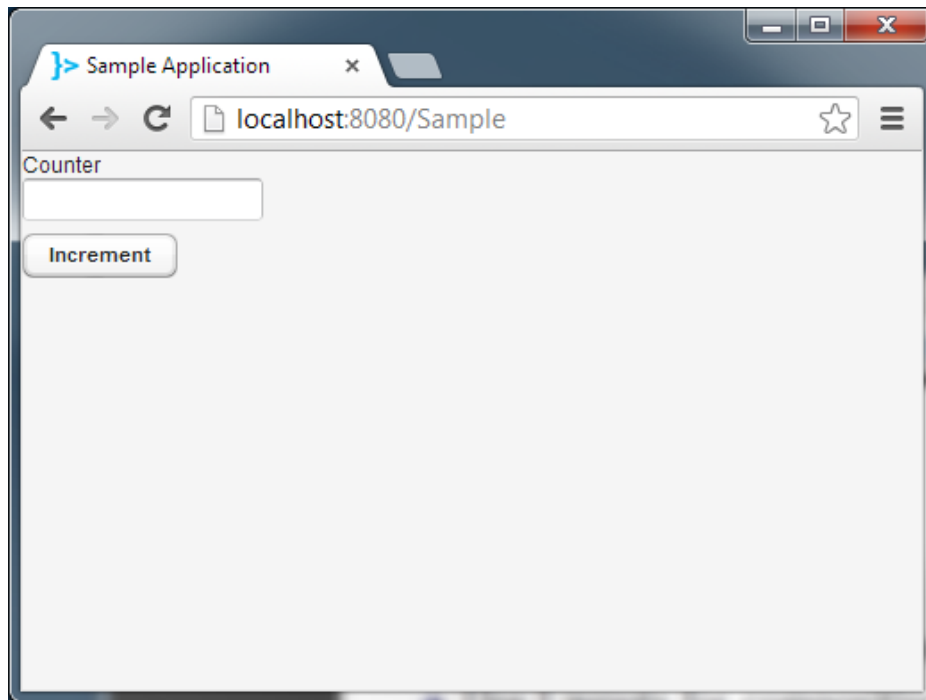
    @Override
    public void init() {
        setMainWindow(new Window("Sample Application"));

        textField = new TextField("Counter");
        button = new Button("Increment");

        button.addListener(new Button.ClickListener() {
            public void buttonClick(ClickEvent event) {
                textField.setValue(++counter);
            }
        });

        // layout
        VerticalLayout layout = new VerticalLayout();
        layout.setSpacing(true);
        layout.addComponent(textField);
        layout.addComponent(button);
        getMainWindow().setContent(layout);
    }
}
```

Laden der Applikation



1. Lade Seite

HTML Skelett
JavaScript (GWT)
Resources (Images,CSS)

→ 830k



2. Lade UI Beschreibung via Ajax

UIDL?repaintAll=1&sh=1200...

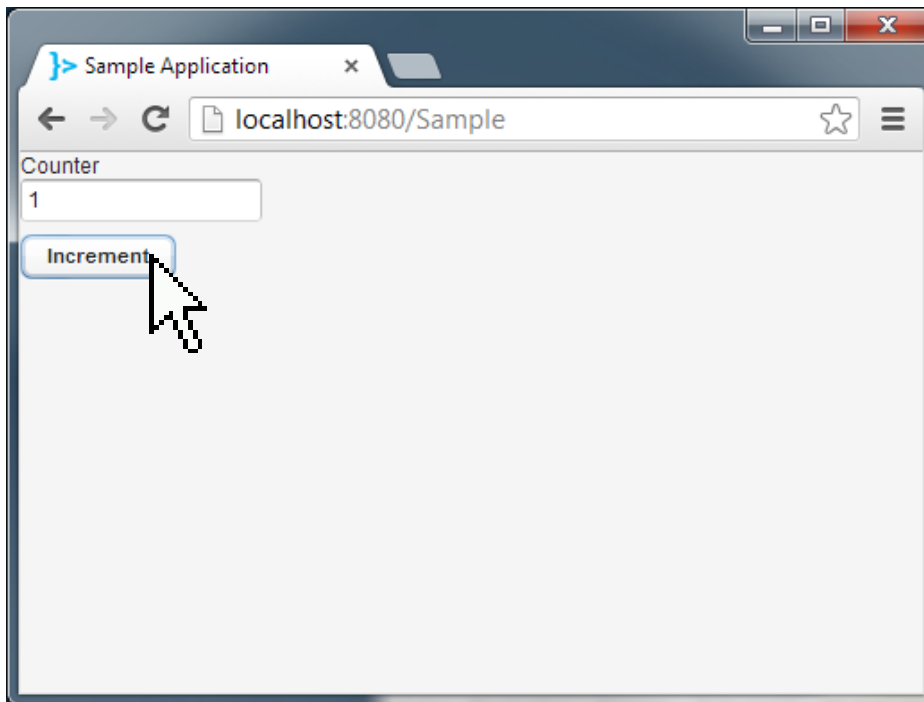
<...json...>

→ 1.3k



3. Rendern der Widgets (js)

Interaktion



1. Button Klick: XHR POST

UIDL?windowName=1

Daten:

ef2024db-ac97-47ca-ac62-
f6dc084f4955435PID0...

↓ Changes

```

▼ for(;;);[{changes:[[change, {format:uid1, pid:PID0}, [0,...]], meta:{}, res
▼ 0: {changes:[[change, {format:uid1, pid:PID0}, [0,...]], meta:{}, resourc
  ▼ changes: [[change, {format:uid1, pid:PID0}, [0,...]],...]
    ▼ 0: [change, {format:uid1, pid:PID0}, [0,...]]
      0: "change"
      ▶ 1: {format:uid1, pid:PID0}
      ▶ 2: [0,...]
    ▼ 1: [change, {format:uid1, pid:PID2}, [2, {id:PID2, caption:Counter, v:{
      0: "change"
      ▶ 1: {format:uid1, pid:PID2}
      ▶ 2: [2, {id:PID2, caption:Counter, v:{text:2}}]]
    locales: []
    meta: {}
    resources: {}
    ▶ timings: [13, 11]
  
```

→ 800b

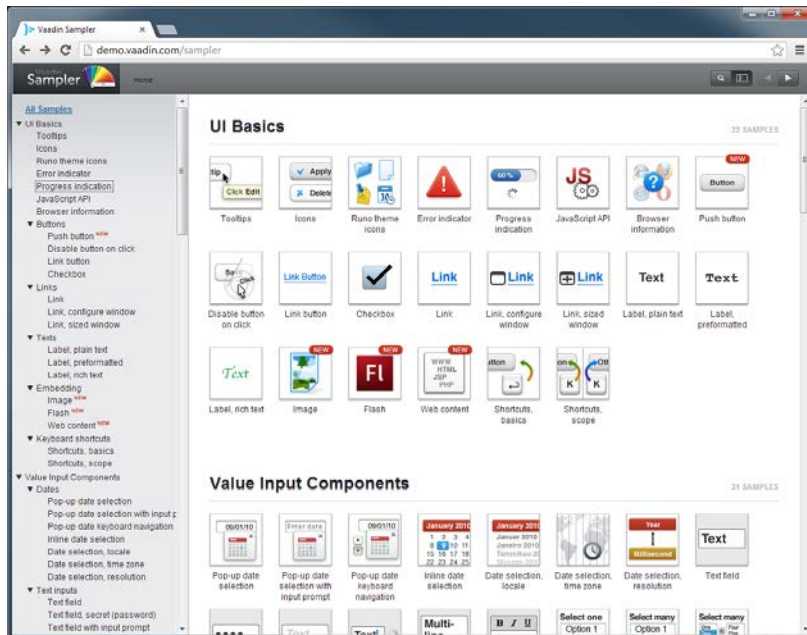
2. Aktualisieren der UI auf Basis von UIDL

UI Layouting

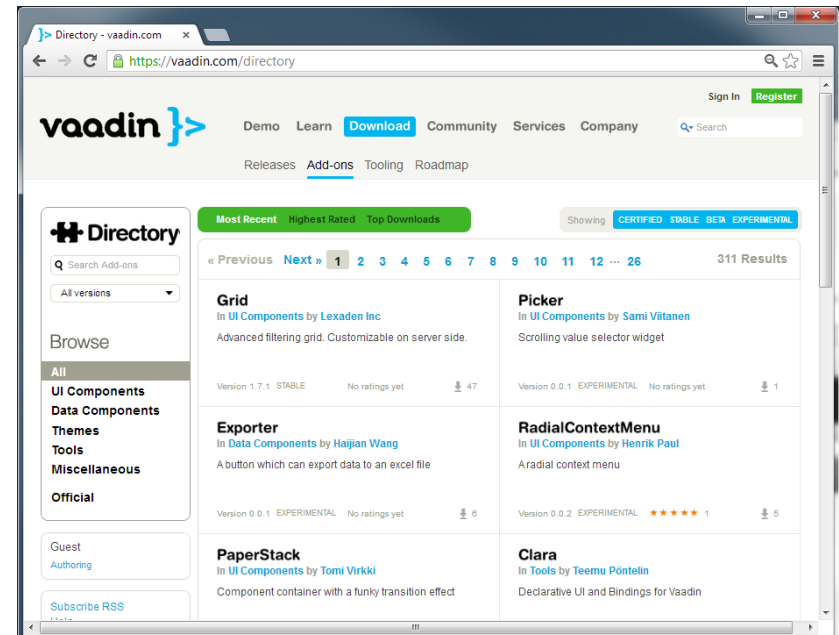
- Fühlt sich an wie in Swing
- Komponenten werden über verschiedene Layouts angeordnet
 - Vertical | Horizontal Layout
 - GridLayout
 - FormLayout
 - CssLayout
 - AbsoluteLayout
 - ...
- Sub-Windows

UI Komponenten

Vaadin Bibliothek

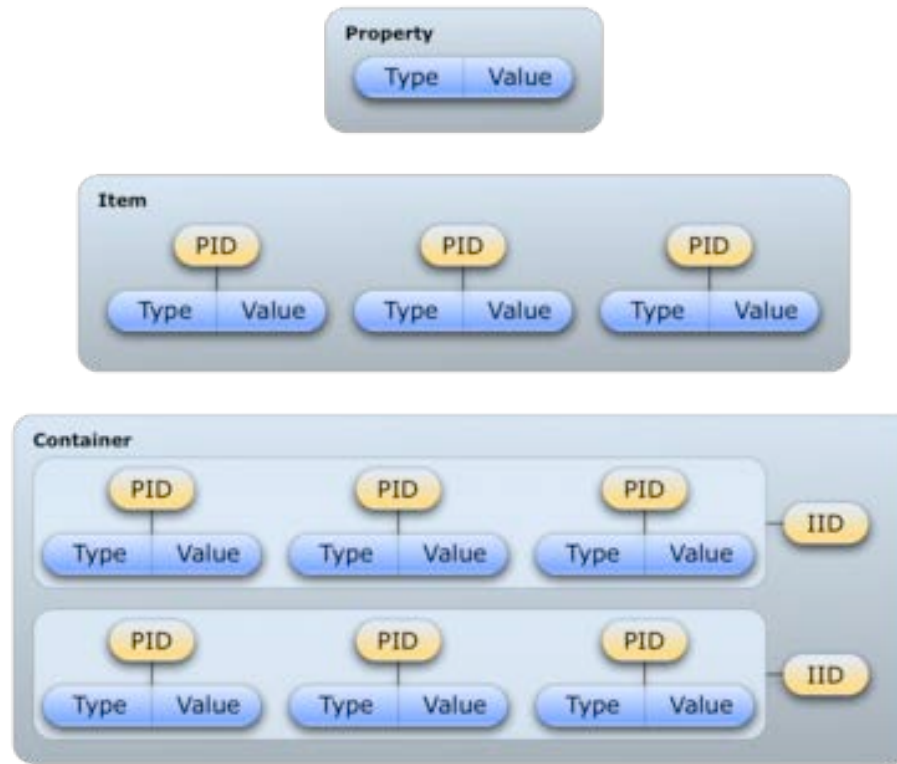


Plugins Directory

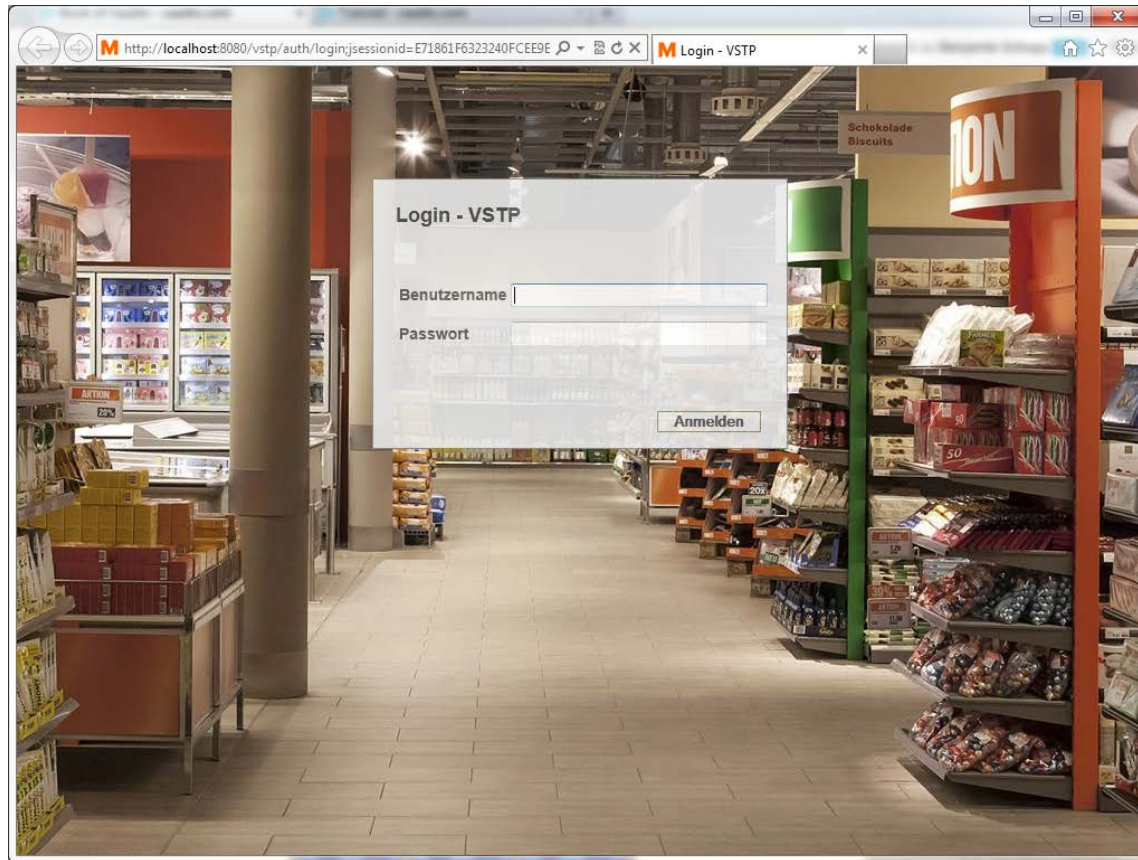


URL: <http://demo.vaadin.com/sampler>

Datenmodell



Projektvorstellung



Projektliste



[Projekte](#)
[Administration](#)
[Einstellungen](#)
[Downloads](#)
[Abmelden](#)
[?](#)

Admin Tester [de]

Projekt auswählen

Name, Beschreibung

PROJEKTNAME	GM	ERSTELLT AM	AKTUALISIERT	ERSTELLT VON	STATUS	TYPE	FFDB	STAMMDATEN	BESCHREIBUNG
Binz	GE	20.12.2012		Tester, Admin	Planung	Neubau	20.12.2012	20.12.2012	
Zürich Nord	BS	20.12.2012		Tester, Admin	Planung	Neubau	20.12.2012	20.12.2012	
Bern Bümpliz	AA	04.01.2013		Tester, User	Planung	Neubau	20.12.2012	20.12.2012	
Zürich West	ZH	07.01.2013		Tester, Admin	Aktiv	Neubau	20.12.2012	20.12.2012	

Projektliste

View

```
// init searchField
ClearableSearchField searchField =
    new ClearableSearchField(
        getI18N().getMessage("button.filter"),
        getI18N().getMessage("button.delete"));

// bind click event
getEventBinder().addListener(searchField,
    "valueChange", "filterSearch");

BeanItemContainer<Project> dataSource = getItemContainer();

// init table
Table projectsTable = new Table();
projectsTable.setContainerDataSource(dataSource);
projectsTable.setVisibleColumns(
    new String[]{"name", "gm", "created", "..."});
projectsTable.setSelectable(true);
projectsTable.setImmediate(true);
projectsTable.setPageLength(0);
```

Presenter

```
// event handler
public void onFilterSearch(ValueChangeEvent event) {

    String searchString = view.getSearchValue();

    projectsContainer.removeAllItems();

    List<Project> projects =
        projectRepository.find(searchString);

    projectsContainer.addAll(projects);

}
```

Kontextmenu

View

```
final Action ACTION_MANAGE = new Action(getI18N().getMessage("action.manage"));
final Action ACTION_DELETE = new Action(getI18N().getMessage("action.delete"));

projectsTable.addActionHandler(new Action.Handler() {

    public Action[] getActions(Object target, Object sender) {
        return new Action[]{ ACTION_MANAGE, ACTION_DELETE };
    }

    public void handleAction(Action action, Object sender, final Object target) {
        if (ACTION_MANAGE == action) {
            getEventBus().manageProject(((Project)target).getId());
        } else if (ACTION_DELETE == action) {
            getEventBus().deleteProject(((Project)target).getId());
        }
    }
});
```

TreeTable

- Bedenken wegen Geschwindigkeit → Prototyp

15 Spalten, 2000 Einträge, ~Styling

Planetary Bodies

name	name1	name2	name3	parent	myname0	myname1	myname2	myname3	myname4	myname5
Item 145	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 146	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 147	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 148	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 149	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 150	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 151	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 152	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 153	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 154	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 155	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 156	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 157	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 158	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1
Item 159	prop 1	prop 2	prop 3	ch.mptei	prop 1	prop 1	prop 1	prop 1	prop 1	prop 1

TreeTable

	BW	BB	VS	FFK	FFK	ST/Anz.	Plan Blfm	Ums	Prod	Δ BB Blfm	Ums	FFK	ST/Anz.	Tool Blfm	Ums	Prod
▼ Food							281.38	14'397'086	51'165	-39.67	3'295'533			281.38	14'397'086	51'165
▼ KOLONIAL	01						281.38	14'397'086	51'165	-39.67	3'295'533			281.38	14'397'086	51'165
<i>Aktion 70er Food</i>	01					16	11.20							11.20		
▼ SUESSWAREN	01	01					23.00	2'133'424	92'757	-4.54	36'188			23.00	2'133'424	92'757
CHOCO SNACKS	01	01	1001		H	005 6.00	6.00	372'000	62'000			H	005	6.00	372'000	62'000
EVENTS	01	01	1002		H	001 1.90	1.90	11'400	6'000			H	001	1.90	11'400	6'000
PRALINES	01	01	1003		H	005 4.00	4.00	204'000	51'000			H	005	4.00	204'000	51'000
TAFELSCHOKOLADE	01	01	1004		H	005 7.00	7.00	504'000	72'000			H	005	7.00	504'000	72'000
OSTERN	01	01	1006		H	001 2.00	2.00	310'874	155'437			H	001	2.00	310'874	155'437
BONBONS	01	01	1008		H	005 5.60	5.60	212'800	38'000			H	005	5.60	212'800	38'000
WEIHNACHTEN	01	01	1009		H	001 2.00	2.00	288'766	144'383			H	001	2.00	288'766	144'383
KAUGUMMI	01	01	1010		H	005 0.40	0.40	37'600	94'000			H	005	0.40	37'600	94'000
SELLOUT	01	01	1011		H							H				
KASSENZONE	01	01	1029		H	001 2.00	2.00	191'664	95'832			H	001	2.00	191'664	95'832
SCANNING KOLONIAL	01	01	1034		H							H				
ALLERGIKER SORTIMENT	01	01	1357		H	005 0.04	0.04	320	8'000			H	005	0.04	320	8'000
▶ BISCUITS	01	02					13.00	981'162	75'474	-0.53	486'687			13.00	981'162	75'474
▶ FRUEHSTUECK/WARMGETRÄNKE	01	03					35.00	2'069'360	59'124	-1.53	592'455			35.00	2'069'360	59'124
▶ BACKSHOP	01	04					13.60	918'560	67'541	-1.84	274'014			13.60	918'560	67'541
▶ TK-HOUSE	01	05					51.58	1'719'160	33'329	7.04	368'827			51.58	1'719'160	33'329
▶ GETRAENKE/GESUNDHEIT	01	07					47.20	1'760'600	37'300	-11.80	159'439			47.20	1'760'600	37'300
▶ BEILAGEN/ZUTATEN	01	10					39.00	2'014'640	51'657	-15.31	403'611			39.00	2'014'640	51'657
▶ KONSERVEN/METHNIC	01	11					25.00	1'400'240	56'009	-7.22	445'226			25.00	1'400'240	56'009
▶ APERO	01	16					14.00	786'440	56'174	-0.04	375'651			14.00	786'440	56'174
▶ TIERWELT	01	18					20.00	613'500	30'675	-3.90	154'421			20.00	613'500	30'675

TreeTable Geschwindigkeit

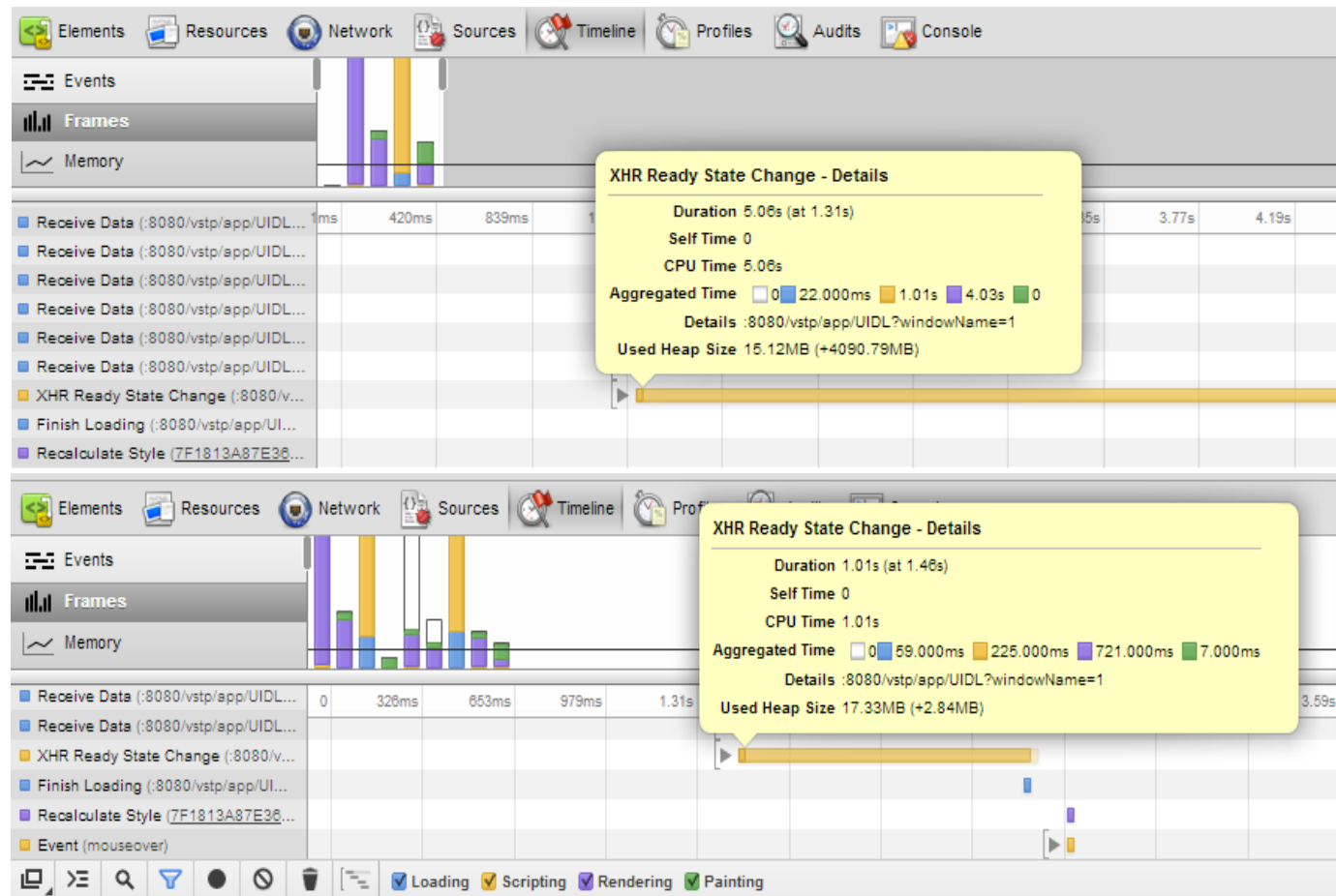
Vorher:

4.06 sec



Nachher:

721ms



CustomLayout

Html

```
<div class="tooltip">
  <span location="1"></span>
  <span location="2"></span>
  <span location="3"></span>
  <span location="4"></span>
  <span location="5"></span>
</div>
```

Java

```
CustomLayout layout = new CustomLayout(templateStream);

layout.addComponent(label1, "1");
layout.addComponent(textField1, "2");
```

Vergleich: HorizontalLayout

```
HorizontalLayout layout = new HorizontalLayout ();

layout.addComponent(label1);
layout.addComponent(textField1);
```

HorizontalLayout / CustomLayout

CustomLayout

```
<div style="border: none; margin: 0px; padding: 0px; width: 280px;" class="v-customlayout">
  <div class="tooltip">
    <span location="1" >
      <div class="v-label ..." >B</div>
    </div>
  </div>
```

Horizontal Layout

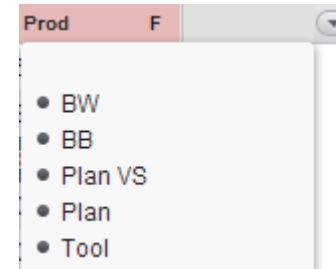
```
<div style="overflow: hidden; width: 225px; height: 19px;" class="horizontallayout">
  <div style="width: 225px; height: 19px; ..." >
    <div style="height: 19px; width: 50px; ..." >
      <div style="float: left; margin-left: 0px;" >
        <div class="v-label v-label-inline5 inline5" style="width: 50px;" >B</div>
      </div>
    </div>
  </div>
```

~ 50 x pro Zeile

~ 1500 x im Screen

UI Herausforderungen: Table

Tool					Referenz (Limmatplatz)				
Blfm	m ²	Ums	Prod	F	Blfm	m ²	Ums	Prod	F
944.46	2'627	50'000'028	52'940	2.78	1'138.95	3'021	46'820'659	41'108	2.65
253.68	634	13'614'327	53'666	2.50	288.13	621	10'341'225	35'890	2.16
203.54	924	25'915'701	127'326	4.54	203.71	1'011	24'314'596	119'358	4.97
377.39	886	9'120'903	24'168	2.35	580.28	1'245	11'013'218	18'979	2.15
109.85	182	1'349'096	12'281	1.66	66.83	143	1'151'618	17'232	2.15



- Mehrere Spalten zusammenfassen für gemeinsames Drag and Drop
- Farbige/Mehrzeilige Header
-> Anpassungen an GWT Clientcode

UI Herausforderungen

- Scrolling / Scrollbars
- Browser Back / Bookmarks (#fragment)
- Multiple Browser Windows
- Browser Performance (IE7,IE8)

Verwendete Vaadin Add-Ons

- I18n
- Refresher
- ConfirmDialog
- EasyUploads
- Notifique

Erkenntnisse

- Stabiles Framework mit vielen guten Add-Ons
- Schnelle Entwicklung (vor allem mit JRebel)
- Layouting: Vorsicht Falle, nicht alles möglich
- Anpassungen vom GWT-Client Code zeitaufwändig
- Langsames Netzwerk beeinträchtigt Usability

Mit Vaadin

- Versprechungen / Erwartungen

Reifes Framework mit aktiver Community und Doku ✓

Solide Komponentenbibliothek und Add-ons ✓

Einfache und schnelle Entwicklung! ✓ !

- Kritische Punkte

- Skalierbarkeit ?

- Hat Probleme bei schlechtem Antwortverhalten? !

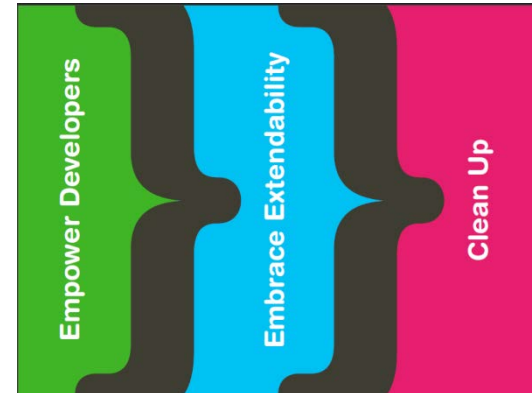
- Komplexität von Entwicklung eigener Komponenten? ✓ !

Ausblick: Vaadin 7

Layout Berechnungen vom Browser

Kein Support mehr für IE 6 und 7
 Verkleinerung des DOM Baumes
 Bessere Performance

- CSS mit Sass
 - Variablen, Funktionen, Vererbung...
- Neue API/Abstraktion für Client-Server Kommunikation
 - Einfachere Erweiterbarkeit
 - Einfacher JavaScript / 3rd Party Widgets zu integrieren
- ...
- ➔ Wird viele der Probleme von Vaadin 6 lösen
- ➔ Mehr Flexibilität beim Einsatz von JS-Frontend-Erweiterungen



Kontakt

Benjamin Schupp schupp@mptechnology.ch

Markus Kinzler kinzler@mptechnology.ch

Offizielle Vaadin Schulungen in Zürich

- 2. – 3. April 2013: Vaadin Fundamentals
- 4. – 5. April 2013: Advanced Vaadin
- Anmeldung über <https://vaadin.com/services#vaadintraining>

