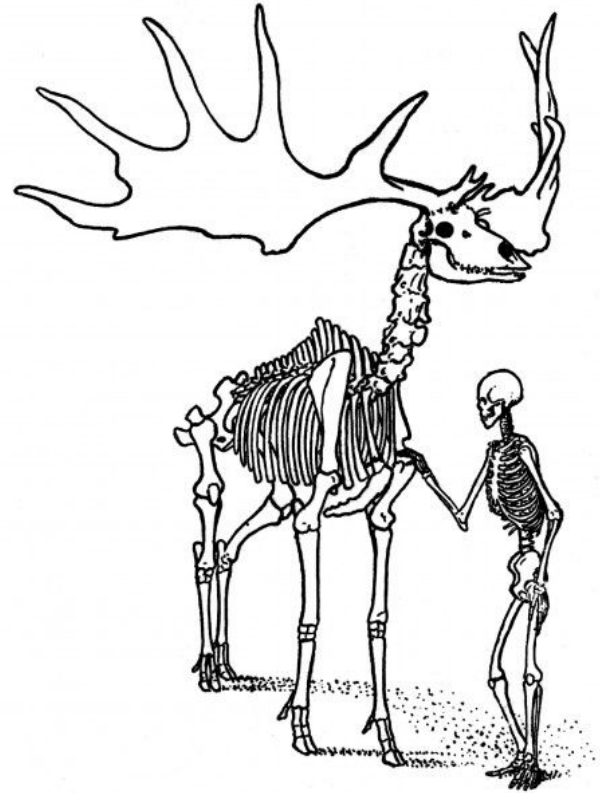


Анатомия и физиология Vaadin Flow

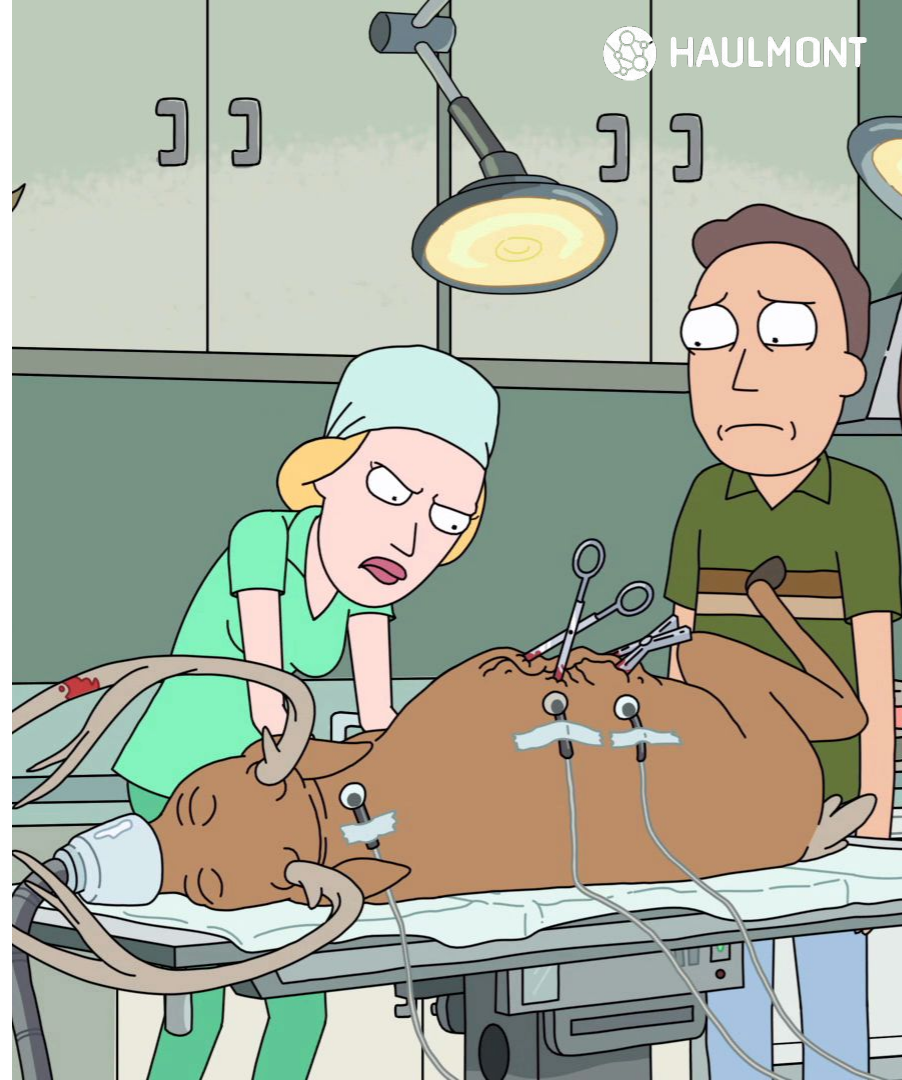


Юрий Артамонов
CUBA Platform



Привет!

Я буду вам всё это
рассказывать, потому что
8+ лет оперирую оленей.



UI на Java - Ужас, Ужас!



Java UI

The image is a collage of three software development environments:

- Top Left:** A database client window showing a table with columns like 'id', 'nom', 'prenom', 'adresse', 'ville', 'codepostal', 'date', 'genre', 'instrument', 'album', 'genre', 'invoice', 'invoiceLine', 'mediaType', 'playlist', 'playlistTrack', 'track', 'views', 'indexes', 'sequences', 'tableTriggers', and 'dataTypes'. The table contains 80 rows of data.
- Top Right:** An IDE window showing a Java class named 'LanguageFolding' with a constructor and a 'buildFoldingDescriptors' method. The code includes annotations like '@NotNull' and '@Nonnull'.
- Bottom:** A map editor window showing a street map with various colored overlays and a sidebar with layers, tags, and selection tools.

Зачем UI на Java?

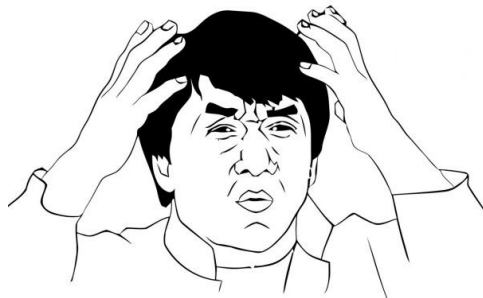
За!

1. На десктопе
2. Для сред разработки
3. Для бизнес приложений
4. Кроссплатформенно



Против!

1. Медленно
2. Нужно много ресурсов
3. Мало наработок по сравнению с ...



**UI на сервере -
Ужас, Ужас!**



Сервер-сайд UI и кому это нужно

- Enterprise-приложения
- Административный UI для больших систем
- **Повышенные требования к безопасности**



Плюсы

1. Один язык для всего
2. Проще отлаживать
3. Знакомые инструменты
4. Быстрее разработка
5. Не нужны веб-сервисы

Минусы

1. Ограниченное масштабирование
2. Сложнее кастомизация
3. Требовательнее к ресурсам





История Vaadin

Основные вехи:

- IT Mill Toolkit 5
- Vaadin 6-8
- Vaadin Flow

```
package com.haulmont.playground;

import ...

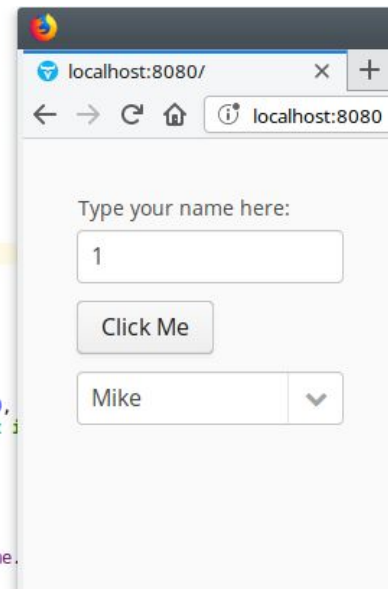
@Theme("mytheme")
public class MyUI extends UI {
    @Override
    protected void init(VaadinRequest vaadinRequest) {
        VerticalLayout layout = new VerticalLayout();

        TextField name = new TextField();
        name.setCaption("Type your name here:");

        User user = new User();
        user.setCount(1);

        new Binder<>(User.class)
            .forField(name)
            .withConverter(new StringToIntegerConverter(0,
                Integer.MAX_VALUE))
            .withValidator(new IntegerRangeValidator("Not a number",
                Integer.MIN_VALUE, Integer.MAX_VALUE))
            .bind("count")
            .read(user);

        Button button = new Button("Click Me");
        button.addClickListener(e ->
            layout.addComponent(new Label("Thanks " + name.getValue()));
    }
}
```



От Google Web Toolkit к фронтенд технологиям



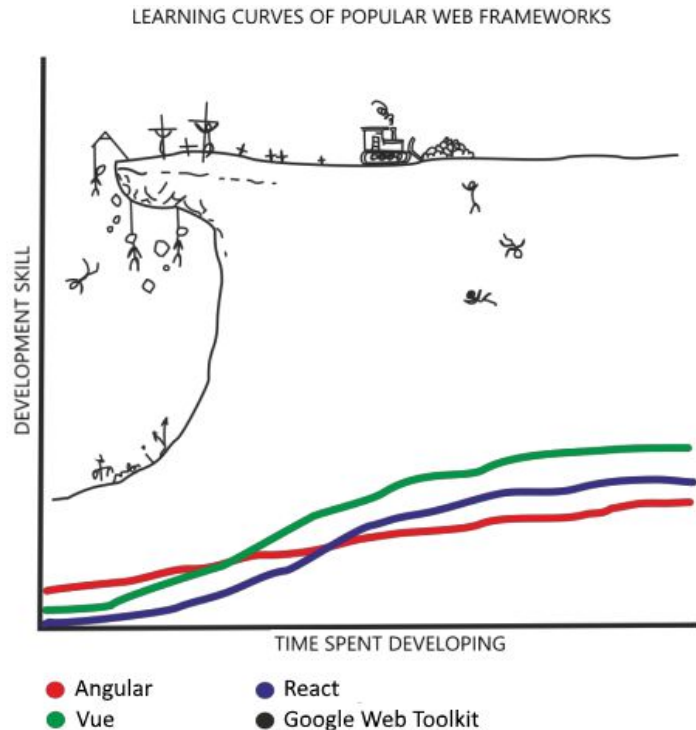
Основные проблемы GWT:

- Устаревшая (немодная) технология
- Медленный и непубличный процесс разработки
- Возрождение JavaScript и появление TypeScript

Куда податься?

Vaadin ≠ GWT

- Сервер-сайт против клиент-сайда
- Нативная Java против эмуляции
- Простота против сложности



Спецификация Web Components

Набор стандартов для создания UI компонентов

Основные концепции:

- Custom Elements
- Shadow DOM
- HTML Template / LitElement
- HTML Imports



<https://www.webcomponents.org/>

0 чём нужно знать перед использованием

- Основное применение - интранет приложения
- Быстро делать типовой UI - сотни однотипных экранов
- Просто начать, сложнее в продакшене
- **Не для веб-сайтов!**



Базовые технологии Vaadin Flow

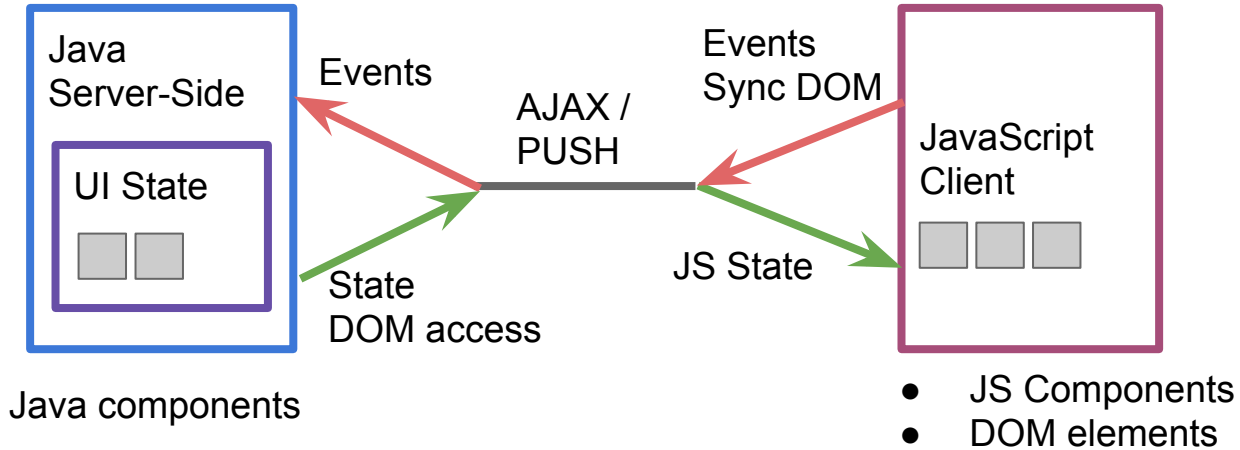


Demo

- Java 8-11 (любой JVM язык)
- Контейнер сервлетов
- Web Components (+ интеграция с Polymer)
- WebJARs для веб-ресурсов
- Добавить Spring Boot по вкусу



Синхронизация состояния UI с сервером



- Двухнаправленная синхронизация данных
- Разница состояний пересылается в JSON
- Поддержка двусторонних вызовов RPC
- Состояние блокируется на момент обработки запроса

Роутинг и управление навигацией

- Декларативный маппинг URL
- Вложенные пути
- Работа с параметрами в URL
- Обработка событий навигации

```
@Route(value = "greet")
public class GreetingComponent extends Div
    implements HasUrlParameter<String> {

    @Override
    public void setParameter(BeforeEvent event, String
parameter) {
        setText(String.format("Hello, %s!", parameter));
    }
}
```

Поддержка WebSocket



Demo

Автоматическая доставка изменений в клиентскую часть.

Режимы:

- Ручной
- Авто WebSocket
- Авто Long-Polling

Под капотом:

Atmosphere Framework

Vaadin Chat

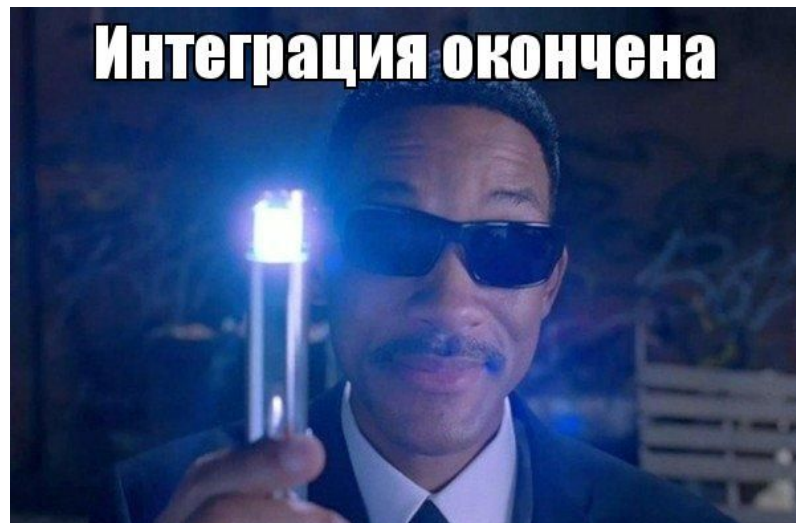
demo: Hello there!

demo2: O hai!

Интеграция Polymer компонентов

Способы интеграции:

- HTML шаблоны
- HTML импорты
- Композитные компоненты

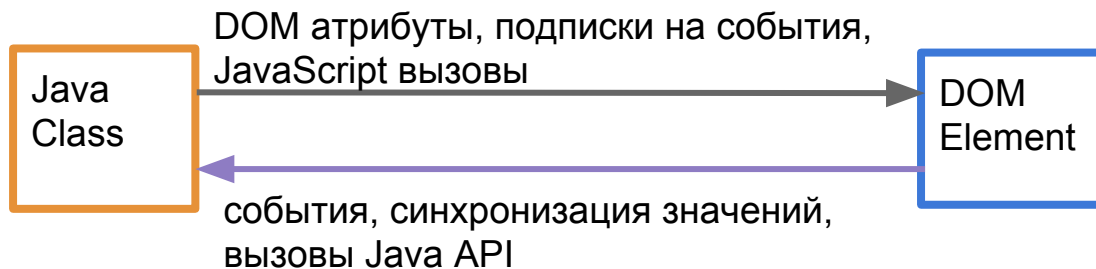


Интеграция JS компонентов



Demo

- Действительно просто:
@JavaScript, @StyleSheet
- Не требуется писать обёртки для JavaScript кода
- Поддержка WebJAR



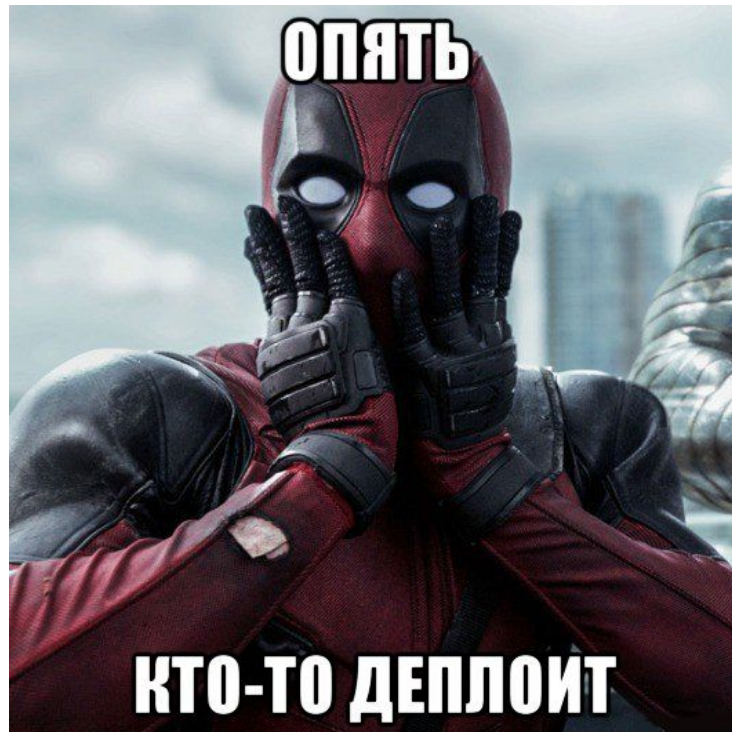
Развёртывание

Пакуем:

- Сборка JAR
- Трансляция JS
- Упаковка веб-ресурсов

Варианты:

- Просто WAR
- Spring Boot



Производительность клиента



Demo

Известные проблемы:

- **Синхронный режим обработки запросов**
- Требуется полное обновление страницы при плохом соединении (при потере пакетов)
- Полифилы WebComponents
- AJAX запрос на сервер для каждого взаимодействия!

Производительность сервера



Demo

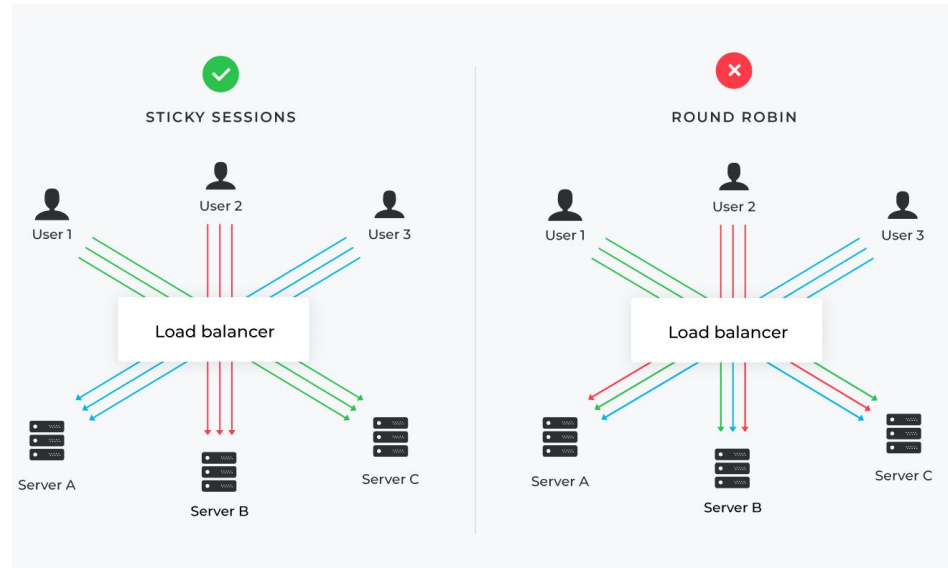
Известные проблемы:

- Блокировка состояния на время обработки запроса
- Повышенные требования к оперативной памяти
- **Проблема синхронизации веб-сессий при горизонтальном масштабировании**

medium.com/jens-jansson/do-vaadin-apps-scale-a0ce4dfefec6

Балансировка нагрузки

- Round robin - не подходит
- **Только sticky sessions**
- Придётся выгнать пользователей из системы после обновления



Интеграция со Spring Boot

vaadin-spring-boot-starter

- **@EnableVaadin**
- **@UIScope**
- **@VaadinSessionScope**



Тут - start.spring.io или vaadin.com/start

Kotlin DSL

Karibu DSL:

1. Меньше кода
2. Пишем на Kotlin
3. Ограниченный набор
компонентов

github.com/mvysny/karibu-dsl

```
@Route("")
class MyView : VerticalLayout() {
    init {
        formLayout {
            textField("Name:")
            textField("Age:")
        }
    }
}
```


Работа в режиме Progressive Web Application

- Базовые атрибуты **@PWA**:
название, иконки, service-worker
- Можно задать заглушку для оффлайн-страницы
- **Нет оффлайн-режима**

Это всё маркетинг

Ну кому-то наверное полезно



Миграция с Vaadin 8

Vaadin 10 и Vaadin 8 похожи:

- Server-side подход
- Java для разработки логики UI
- Одинаковый подход к биндингу данных

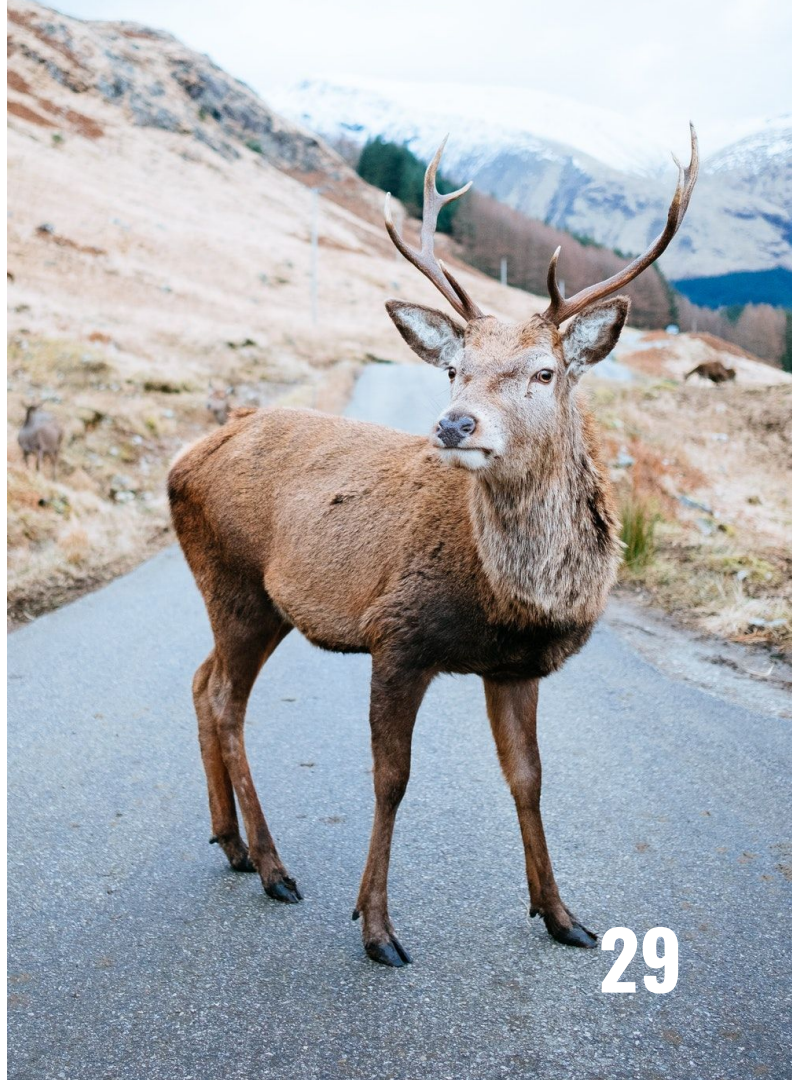


В сухом остатке:

Vaadin 10 это новый фреймворк написанный с нуля.

Выводы

- Vaadin Flow - фронтенд фреймворк со всеми вытекающими
- Рекомендуем **для кровавого энтепрайза**
- Не рекомендуем для публичных веб-приложений
- Можно заменять Vaadin FW



Спасибо!

Вопросы?

Twitter: [@Yuriy_Artamonov](https://twitter.com/Yuriy_Artamonov)

