

Apache Wicket 10 years and beyond

Martijn Dashorst



APACHECON
EUROPE

CORINTHIA HOTEL
BUDAPEST, HUNGARY
— NOVEMBER 17-21, 2014 —



a brief history
of Wicket

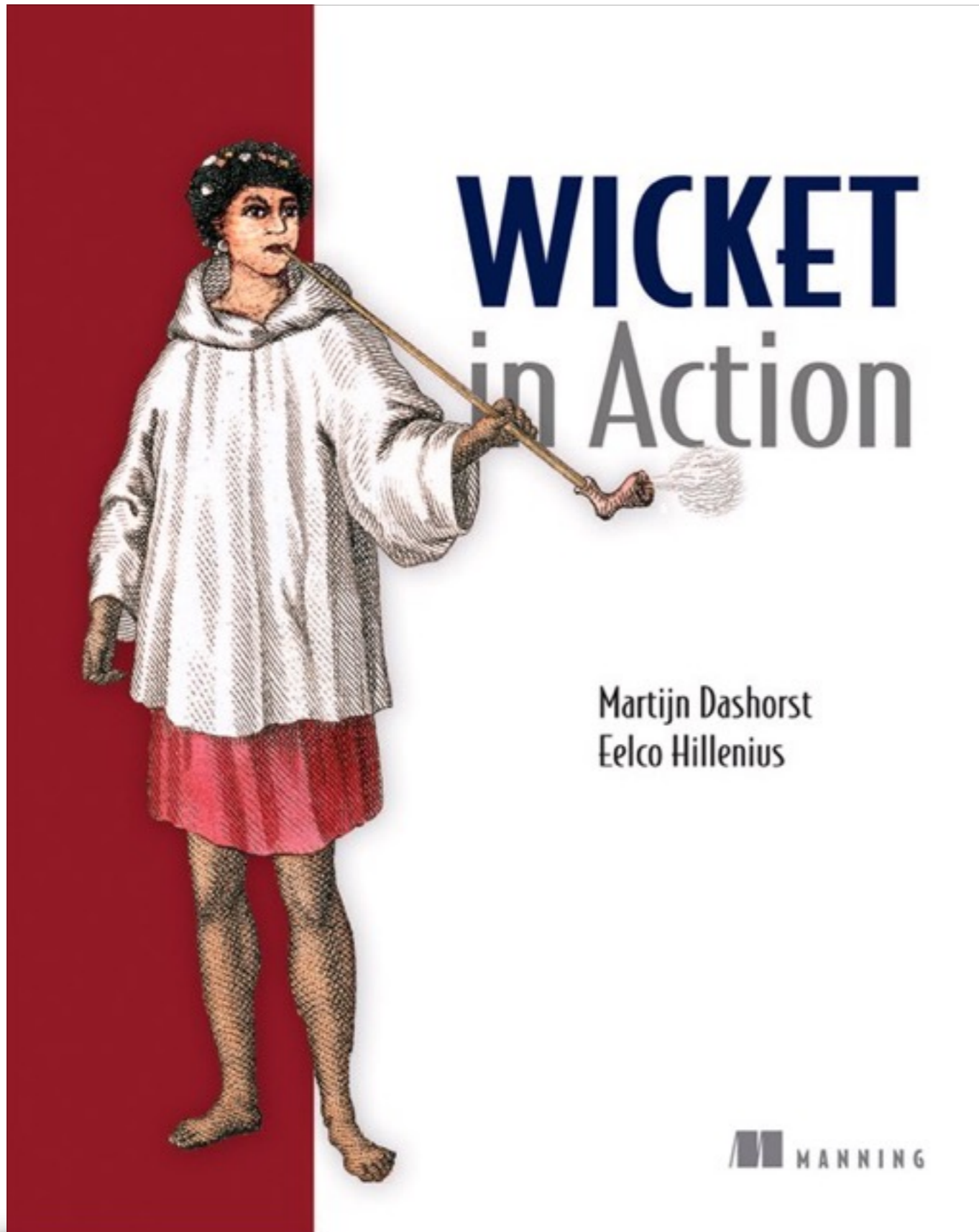


the current state
of Wicket



the future
of Wicket





co-author

10 year contributor to **Wicket**

started in 2004 at topicus

without any knowledge of web programming

before Wicket

at topicus



ParnasSys - Microsoft Internet Explorer

stand Beveiken Beeld Favorieten Extra Help

https://start.parnassys.net/bao/main.home.m

ParnasSys Demoschool

Leering Groep Medewerker School Info

Jan Hulst (school administrateur, leerkracht, intern begeleider, school applicatiebeheer)

Home Personalia Aanstelling Tijdregistratie

Home

Verjaardagen

Datum	Leeftijd	Naam	Groep
20 jan	10	Fieke Doornik	6
21 jan	11	Mauro Casarini	8
24 jan	11	Frank Heidotting	8
30 jan	7	Peter van Duin	3
1 feb	11	Robert Aalders	8

Activiteiten

Datum	Categorie	Groep
22 jan	Stuudiemiddag	

Mededelingen

Onderwerp

- Nieuwe overzichten: o.a. overzichten met vier hoofdvakgrafieken
- Vernieuwde overzichten: o.a. Ontwikkelingsperspectief
- NOT gratis kaarten
- Nieuwe toetsnormeringen: R&W2006 en DIG1

Overzichten slechts éénmaal opvragen

ParnasSys - Microsoft Internet Explorer

stand Bewerken Beeld Favorieten Extra Help

https://start.parnassys.net/bao/groups.lstpresentedit.m

ParnasSys Demoschool

Aangemeld als jhulstdemo - Help - Afmelden

Leering Groep Medewerker School Info

Mijn ParnasSys

Groep 8 (2006 / 2007)

Leerlingen Absentie Rooster Toetsen Rapporten Zien! Begeleiding Groepsplannen Map Gegevens

Absentie

week 3 - 2007

	Ma (15 jan)		Di (16 jan)		Wo (17 jan)		Do (18 jan)		Vr (19 jan)		
	Ocht	Mid	Ocht	Mid	Ocht	Mid	Ocht	Mid	Ocht	Mid	
Robert Aalders											
Lena van Ark		Z	Z								
Boris J van Basten											
Kaylee Bosscher											
Vanessa Botha											
Mauro Casarini											
Ellen van Dijk											
Frank Heidotting											
Cilius Hingst											
Trudy Horst											
Lydia Karels											
Cathy Praalder								Z	Z	Z	Z
Jeanine Rehorst											

Betekenis afkortingen:

- BG = Bus Gemist
- LL = Leesles Taalkliniek
- TL = Te laat
- Ta = Tandarts
- S. = Spijbelen
- BG = Begravenis
- STT = Stottertherapie
- BL = Bruiloft
- Z = Ziek
- vkg = vak ongeoorloofd
- ? = Onbekend
- D. = Dokterbezoek
- V. = Vakantie

Opslaan en naar overzicht

ParnasSys

2004

using maverick velocity
hibernate jasperreports

lots of XML configuration

1 change requires N files to be modified

(for large values of N)

```
<command name="admin.tabelbeheer.edittoetsonderdeel">
  <controller class="nl.topicus.bao.web.ctrl.lab.toetsen.
  <view name="success" path="/admin/tabelbeheer/toetsen/e
    <transform path="/mlayout/mainlayout.vm">
      <param name="_pagetitle" value="Beheer Tabellen
    </transform>
  </view>
  <view name="error" path="/admin/tabelbeheer/toetsen/edi
    <transform path="/mlayout/mainlayout.vm">
      <param name="_pagetitle" value="Beheer Tabellen
    </transform>
  </view>
  <view name="list" path="admin.tabelbeheer.edittoets.m"
</command>
```


high learning curve

no back button support
no multi-tab support

HTTP Session dumping
ground

little to no reuse

```
</tr>
#if($model.bean.absentieRegels.empty)
    <tr>
        <td colspan="19" class="border-r">#formname('groep.absent'
    </tr>
#else
#set($columnIndexes=[1,2,3,4,5,6,7,8,9,10])
#foreach($row in $model.bean.absentieRegels)
    <tr>
        <td class="columnvalue">$arrayTool.elementAt($row, 0)</td
        #foreach($index in $columnIndexes)
            #set($melding=$arrayTool.elementAt($row, $index))
```

complex UI neigh impossible

Vocus Versie V1.11 - Joop - Vocus Demo School [help](#)

[Leerling](#) [Groep](#) [Medewerker](#) [Onderwijs](#) [Financieel](#) [Beheer](#) [Info](#) Lnr:

105219 - Cilla Achtereekte L4bzw2 ✕

[Leerlingkaart](#) [Personalia](#) [Registratie](#) [Plaatsing](#) [Vakken](#) [Absentie](#) [Resultaten](#) [Examen](#) [BRON](#) [Begeleiding](#)

Resultaten

[D1d](#) [S2g](#) [S2g](#) [L3bzw2](#) [L4bzw2](#) **Examendossier**

Opleiding: Basisbg Zorg en Welzijn ZW Lichting: 2005/2007

Sorteren op: [Links uitlijnen](#) Toon historische vakkeuzes (0)

Vak									TT	PO	SE
Nederlandse taal	6.7	6.0	5.8	5.5	5.7						5.9
Engelse taal	7.3	6.5	7.2	7.5	4.2						6.5
kunsvakken I (gemeenschappelijk deel)	6.0	6.7									6.4
maatschappijleer	5.8	7.5	5.6								6.3
lichamelijke opvoeding	7.0	7.0	7.0	7.0	6.5						6.9
biologie	6.5	5.1	5.1	5.7	5.6						5.6
maatschappijleer II	7.2			5.7							6.4
levensbeschouwing	7.0										7.0
mentorles											-
orientatie studie en beroep											
zorg en welzijn breed	4.9	6.5	5.6	4.7	7.0						5.7

a framework that makes
**reuse possible, minimises
configuration, server side
state management** easy
and is as **type safe** as
possible

Action Framework Anvil Apache Click Apache MyFaces Apache
Shale Apache Sling Apache Struts Apache Struts 2 Apache
Tapestry Apache Turbine Apache Wicket AppFuse Aranea Web
Framework AribaWeb Aurora Baritus Barracuda Bento Bishop Brill
Calyxo Cameleon Canyamo Caramba Cassandra Chiba Chrysalis
Dinamica Dovetail DWR Echo Eclipse RAP Espresso flexive Flower
framework Folium FormEngine Genie Grails GWT Hamlets Helma
Induction ItsNat Jacquard Jaffa Japple JAT JATO JBanana JBoss
Seam Jeenius JFormular JOSSO JPublish JSPWidget Jspx-bay
jStatemachine Jucas JVx JWAA JWarp jWic jZeno jZonic Macaw
Makumba Maverick Melati Mentawai Millstone Nacho Niggle
OpenEmcee OpenLaszlo OpenXava Oracle ADF OXF Pandora
Playframework Pustefix Restlet RIFE Roma Meta Framework RSF
Scope SerfJ Shocks Smile SOFIA Sombrero Spark Spring MVC
Strecks Stripes Swinglets SwingWeb Tapestry TeaServlet ThinWire
Trimpath Junction Turbine Vaadin Verge VRaptor Vroom Warfare
Wavemaker WebObjects WebOnSwing WebWork wingS Xoplon Ze
Framework ZK ztemplates

120+ java web frameworks

Sponsored Links

Implementing Sun Microsystems' Core J2EE Patterns [white paper](#), sponsored by Compuware

[Download](#) Gluecode's Portal and Business Process Management software

[Download](#) IBM Websphere Express V 5.1.1 with production Java Server Faces Support

[JAOO 2004 Conference](#) on Development, Methods, Best Practices. Sept 20-24, Denmark

News forum

Messages: 86

[Bookmark thread](#)

[Printer friendly](#)

[Post reply](#)

[XML](#)

Wicket: Do we need yet another presentation layer framework?

Posted By: [Miko Matsumura](#) on August 18, 2004 @ 02:03 PM

The answer could be yes. The new [Wicket](#) project at [CodeHaus](#), written by a former member of the Swing UI engineering team at Sun Microsystems, takes a unique and simple approach to web UIs.

Like Tapestry, Wicket decorates HTML tags with a special name attribute. But unlike Tapestry, it is dirt simple, having no further HTML syntax, no XML configuration files and a much simpler Swing-like component model.

Wicket was also designed for easy Hibernate integration. With an IDE plugin or two, a good basic idea like this could just turn out to be a better idea than JSP or JSF, especially in a world moving away from legacy J2EE practices. Wicket is Open Source under the Apache Software License.

Visit the [Wicket home page](#)

Threaded replies

- [Wicket: Do we need yet another presentation layer framework?](#) by [Miko Matsumura](#) on 08/18, 02:03 PM
- [Wicket: Do we need yet another presentation layer framework?](#) by [Mark Nuttall](#) on 08/18, 02:26 PM
 - [Echo](#) by [Jonathan Locke](#) on 08/18, 02:53 PM
 - [Echo](#) by [Mark Nuttall](#) on 08/18, 02:59 PM
 - [Echo](#) by [Mark Nuttall](#) on 08/18, 03:02 PM
 - [Differences](#) by [Jonathan Locke](#) on 08/18, 03:23 PM
 - [Differences](#) by [Mark Nuttall](#) on 08/18, 03:44 PM
 - [Yes](#) by [Jonathan Locke](#) on 08/18, 03:49 PM
 - [Yes](#) by [Mark Nuttall](#) on 08/18, 03:54 PM
 - [Yes](#) by [Jonathan Locke](#) on 08/18, 04:03 PM
 - [Try EchoStudio](#) by [Michael Imari](#) on 08/18, 03:52 PM
 - [IDE Pluginst](#) by [Jonathan Locke](#) on 08/18, 03:58 PM
 - [HtmlTemplatePanels](#) by [Jonathan Locke](#) on 08/18, 03:43 PM
 - [And...](#) by [Jonathan Locke](#) on 08/18, 03:45 PM
- [Wicket: Do we need yet another presentation layer framework?](#) by [Tom Davies](#) on 08/18, 05:46 PM
- [Do we need yet another presentation layer framework? !NOOO!](#) by [sdfdf dgdgg](#) on 08/18, 02:40 PM
- [Do we need yet another presentation layer framework? !NOOO!](#) by [Mark Nuttall](#) on 08/18, 03:03 PM
- [Do we need yet another presentation layer framework? !NOOO!](#) by [Rod Johnson](#) on 08/19, 09:20 AM
- [Do we need yet another presentation layer framework? !YES!](#) by [Archimedes Trajano](#) on 08/19, 11:33 AM
- [I like it so far](#) by [Toby Bevelts](#) on 08/18, 02:57 PM

```
public class EditPage extends WebPage {
    private Person person;

    public EditPage(Person person) {
        this.person = person;

        add(new Button("save") {
            @Override
            public void onSubmit() {
                Entities.save(person);
                setResponsePage(new ShowPage(person));
            }
        });
    }
}
```

a framework that makes

✓ **reuse possible**, ✓ **minimises**

configuration, server side

✓ **state management** easy

and is as ✓ **type safe** as

possible

Action Framework Anvil Apache Click Apache MyFaces Apache Shale Apache Sling Apache Struts Apache Struts 2 Apache Tapestry Apache Turbine Apache Wicket AppFuse Aranea Web Framework AribaWeb Aurora Baritus Barracuda Bento Bishop Brill Calyxo Cameleon Canyamo Caramba Cassandra Chiba Chrysalis Dinamica Dovetail DWR Echo Eclipse RAP Espresso flexive Flower framework Folium FormEngine Genie Grails GWT Hamlets Helma Induction ItsNat Jacquard Jaffa Japple JAT JATO JBanana JBoss Seam Jeenius JFormular JOSSO JPublish JSPWidget Jspx-bay jStatemachine Jucas JVx JWAA JWarp jWic jZeno jZonic Macaw Makumba Maverick Melati Mentawai Millstone Nacho Niggle OpenEmcee OpenLaszlo OpenXava Oracle ADF OXF Pandora Playframework Pustefix Restlet RIFE Roma Meta Framework RSF Scope SerfJ Shocks Smile SOFIA Sombrero Spark Spring MVC Strecks Stripes Swinglets SwingWeb Tapestry TeaServlet ThinWire Trimpath Junction Turbine Vaadin Verge VRaptor Vroom Warfare Wavemaker WebObjects WebOnSwing WebWork wingS Xoplon Ze Framework ZK ztemplates

Action Framework Anvil Apache Click Apache MyFaces Apache
Shale Apache Sling Apache Struts Apache Struts 2 Apache
Tapestry Apache Turbine Apache Wicket AppFuse Aranea Web
Framework AribaWeb Aurora Baritus Barracuda Bento Bishop Brill
Calyxo Cameleon Canyamo Caramba Cassandra Chiba Chrysalis
Dinamica Dovetail DWR Echo Eclipse RAP Espresso flexive Flower
framework Folium FormEngine Genie Grails GWT Hamlets Helma
Induction ItsNat Jacquard Jaffa Japple JAT JATO JBanana JBoss
Seam Jeenius JFormular JOSSO JPublish JSPWidget Jspix-bay
jStatemachine Jucas JvX JWAA JWarp jWic jZeno jZonic Macaw
Makumba Maverick Melati Mentawai Millstone Nacho Niggle
OpenEmcee OpenLaszlo OpenXava Oracle ADF OXF Pandora
Playframework Pustefix Restlet RIFE Roma Meta Framework RSF
Scope SerfJ Shocks Smile SOFIA Sombrero Spark Spring MVC
Strecks Stripes Swinglets SwingWeb Tapestry TeaServlet ThinWire
Trimpath Junction Turbine Vaadin Verge VRaptor Vroom Warfare
Wavemaker WebObjects WebOnSwing WebWork wingS Xoplon Ze
Framework ZK ztemplates

15 java web frameworks alive

i.e. made any release in the last year

including Wicket!

What is Wicket?



Apache Wicket

From Wikipedia, the free encyclopedia

Apache Wicket, commonly referred to as **Wicket**, is a lightweight component-based [web application framework](#) for the [Java programming language](#) conceptually similar to [JavaServer Faces](#) and [Tapestry](#). It was originally written by Jonathan Locke in April 2004. Version 1.0 was released in June 2005. It graduated into an [Apache](#) top-level project in June 2007.^[2]

A brief history of Wicket





News



Discussions



Tech talks



Patterns



Reviews



Articles

Go

410091 mem

Sponsored Links

Implementing Sun Microsystems' Core J2EE Patterns [white paper](#), sponsored by Compuware

[Download](#) Gluecode's Portal and Business Process Management software

[Download](#) IBM Websphere Express V 5.1.1 with production Java Server Faces Support

[JAOO 2004 Conference](#) on Development, Methods, Best Practices. Sept 20-24, Denmark



News forum

Messages: 86

[Bookmark thread](#)[Printer friendly](#)[Post reply](#)[XML](#)

Wicket: Do we need yet another presentation layer framework?

Posted By: [Miko Matsumura](#) on August 18, 2004 @ 02:03 PM

The answer could be yes. The new [Wicket](#) project at [CodeHaus](#), written by a former member of the Swing UI engineering team at Sun Microsystems, takes a unique and simple approach to web UIs.

Like Tapestry, Wicket decorates HTML tags with a special name attribute. But unlike Tapestry, it is dirt simple, having no further HTML syntax, no XML configuration files and a much simpler Swing-like component model.

Wicket was also designed for easy Hibernate integration. With an IDE plugin or two, a good basic idea like this could just turn out to be a better idea than JSP or JSF, especially in a world moving away from legacy J2EE practices. Wicket is Open Source under the Apache Software License.

Visit the [Wicket home page](#)

Threaded replies

- [Wicket: Do we need yet another presentation layer framework?](#) by [Miko Matsumura](#) on 08/18, 02:03 PM
- [Wicket: Do we need yet another presentation layer framework?](#) by [Mark Nuttall](#) on 08/18, 02:26 PM
 - [Echo](#) by [Jonathan Locke](#) on 08/18, 02:53 PM
 - [Echo](#) by [Mark Nuttall](#) on 08/18, 02:59 PM
 - [Echo](#) by [Mark Nuttall](#) on 08/18, 03:02 PM
 - [Differences](#) by [Jonathan Locke](#) on 08/18, 03:23 PM
 - [Differences](#) by [Mark Nuttall](#) on 08/18, 03:44 PM
 - [Yes](#) by [Jonathan Locke](#) on 08/18, 03:49 PM
 - [Yes](#) by [Mark Nuttall](#) on 08/18, 03:54 PM
 - [Yes](#) by [Jonathan Locke](#) on 08/18, 04:03 PM
 - [Try EchoStudio](#) by [Michael Imari](#) on 08/18, 03:52 PM
 - [IDE Pluginst](#) by [Jonathan Locke](#) on 08/18, 03:58 PM
 - [HtmlTemplatePanels](#) by [Jonathan Locke](#) on 08/18, 03:43 PM
 - [And...](#) by [Jonathan Locke](#) on 08/18, 03:45 PM
- [Wicket: Do we need yet another presentation layer framework?](#) by [Tom Davies](#) on 08/18, 05:46 PM
- [Do we need yet another presentation layer framework? !NOOO!](#) by [sdfdf dgdgg](#) on 08/18, 02:40 PM
- [Do we need yet another presentation layer framework? !NOOO!](#) by [Mark Nuttall](#) on 08/18, 03:03 PM
- [Do we need yet another presentation layer framework? !NOOO!](#) by [Rod Johnson](#) on 08/19, 09:20 AM
 - [Do we need yet another presentation layer framework? !YES!](#) by [Archimedes Trajano](#) on 08/19, 11:33 AM
- [I like it so far](#) by [Tohu Revelts](#) on 08/18, 02:57 PM

Links

[News](#)
[CVS](#)
[Roadmap](#)
[Download](#)
[Contact](#)
[Mailing Lists](#)
[Bug Database](#)

Feeds

 [Site](#)
 [News](#)

Home



Welcome to Wicket

Wicket is a Java web application framework that takes simplicity, separation of concerns and ease of development to a whole new level. Wicket pages can be mocked up, previewed and later revised using standard WYSIWYG HTML design tools. Dynamic content processing and form handling is all handled in Java code using a Swing-like component model backed by POJO data beans that can easily be persisted with Hibernate.

- Find out **why you should use wicket**
- View **some examples**
- See **how easy it is to get started**
- Join **a mailing list**
- Download **the SDK**

2004 2005

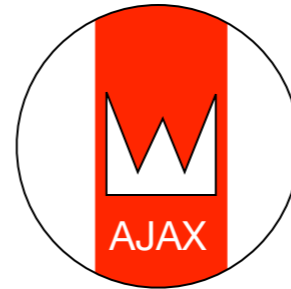
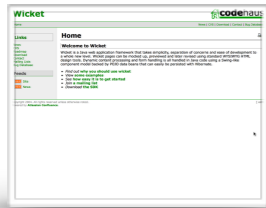


2004 2005



2004 2005

2006



codehaus.org

1.0

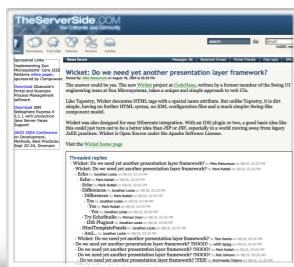
1.2



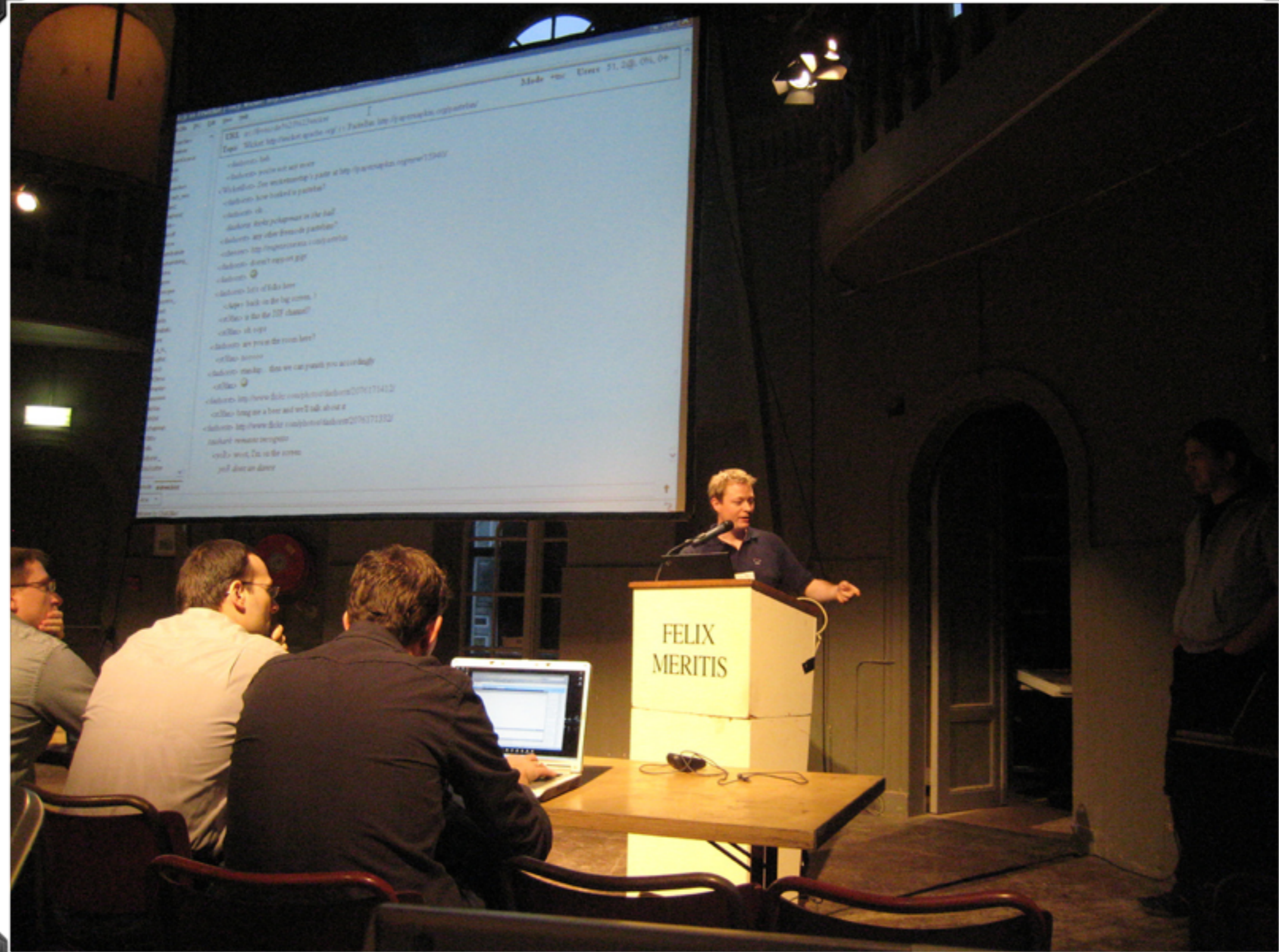
The Server Side

JavaOne

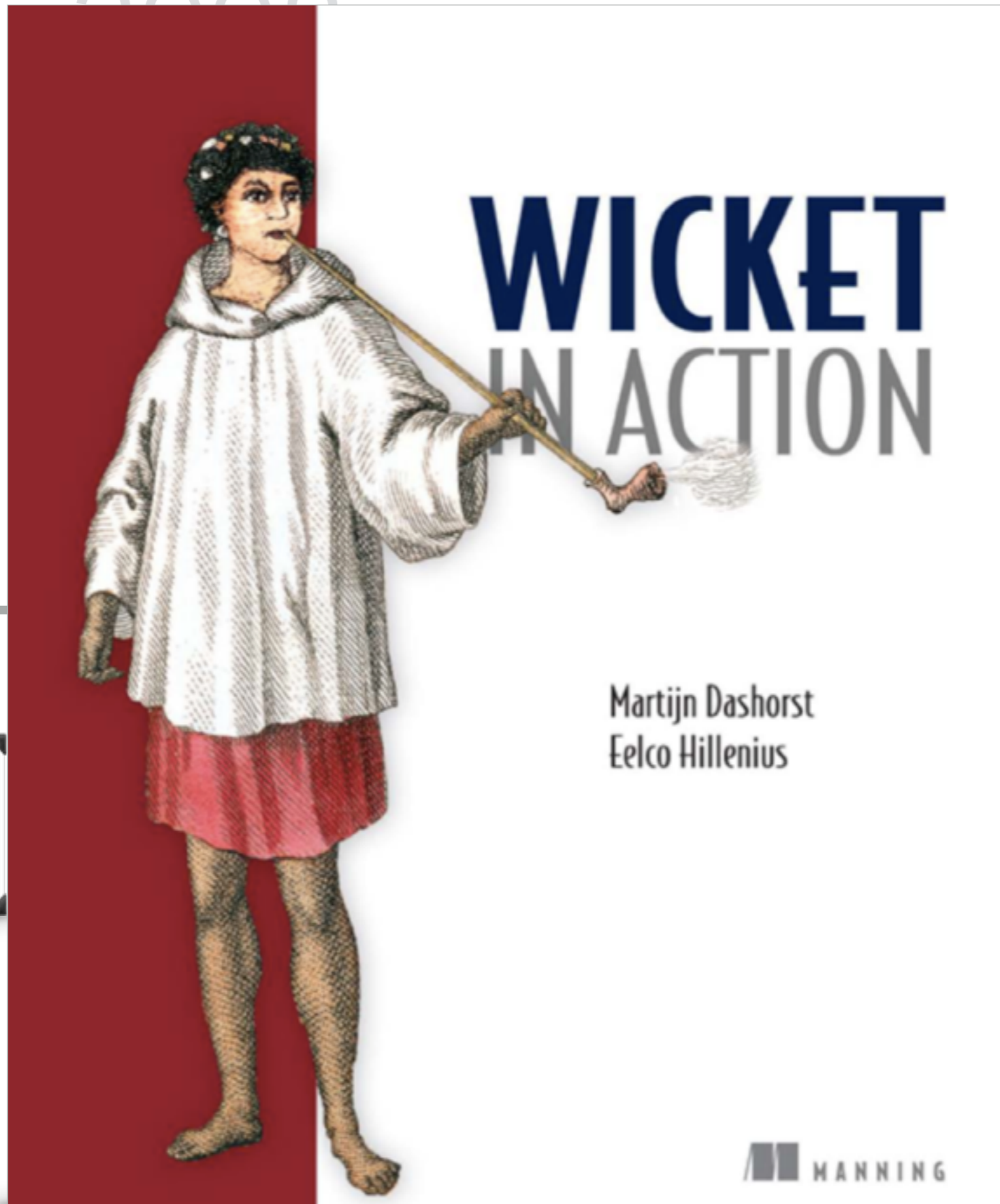
1.1



2007



2007 2000



WICKET IN ACTION

Martijn Dashorst
Eelco Hillenius

 MANNING

2007 2008 2009 2010 2011



1.5
i



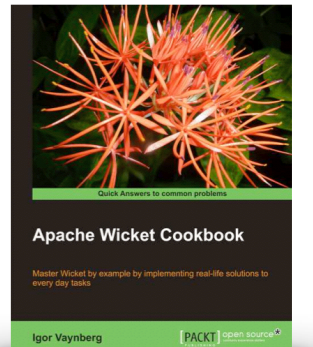
2007 2008 2009 2010 2011



1.3

1.4

1.5



2012 2013

2014

7.0.0-M4

6.2

6.6

6.10

6.14

6.18

7.0.0-M2

6.0

6.4

6.8

6.12

6.16

6.1

6.5

6.9

6.13

6.17

7.0.0-M3

6.3

6.7

6.11

6.15

7.0.0-M1



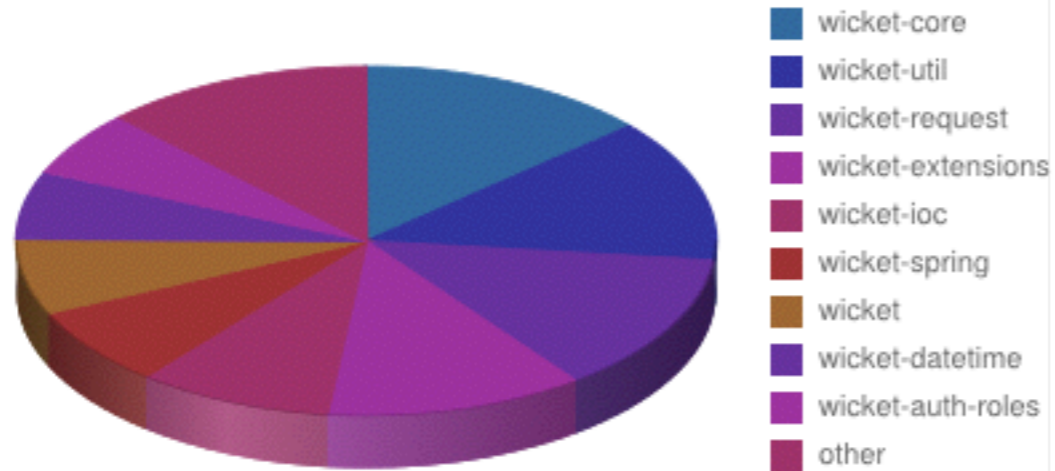
That's **10** years
of Apache Wicket!



The state of Wicket



Downloads From Last 3 Months
For org.apache.wicket



1. core

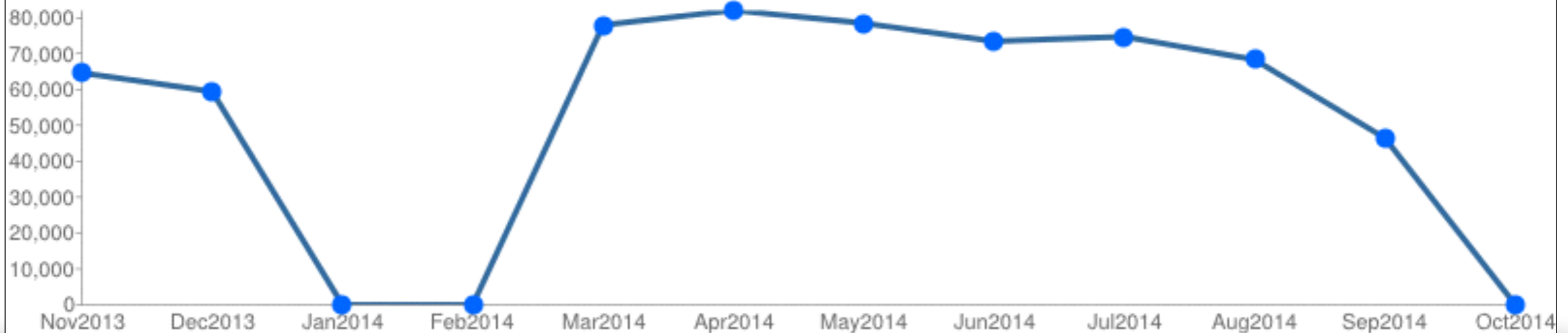
2. extensions

3. spring

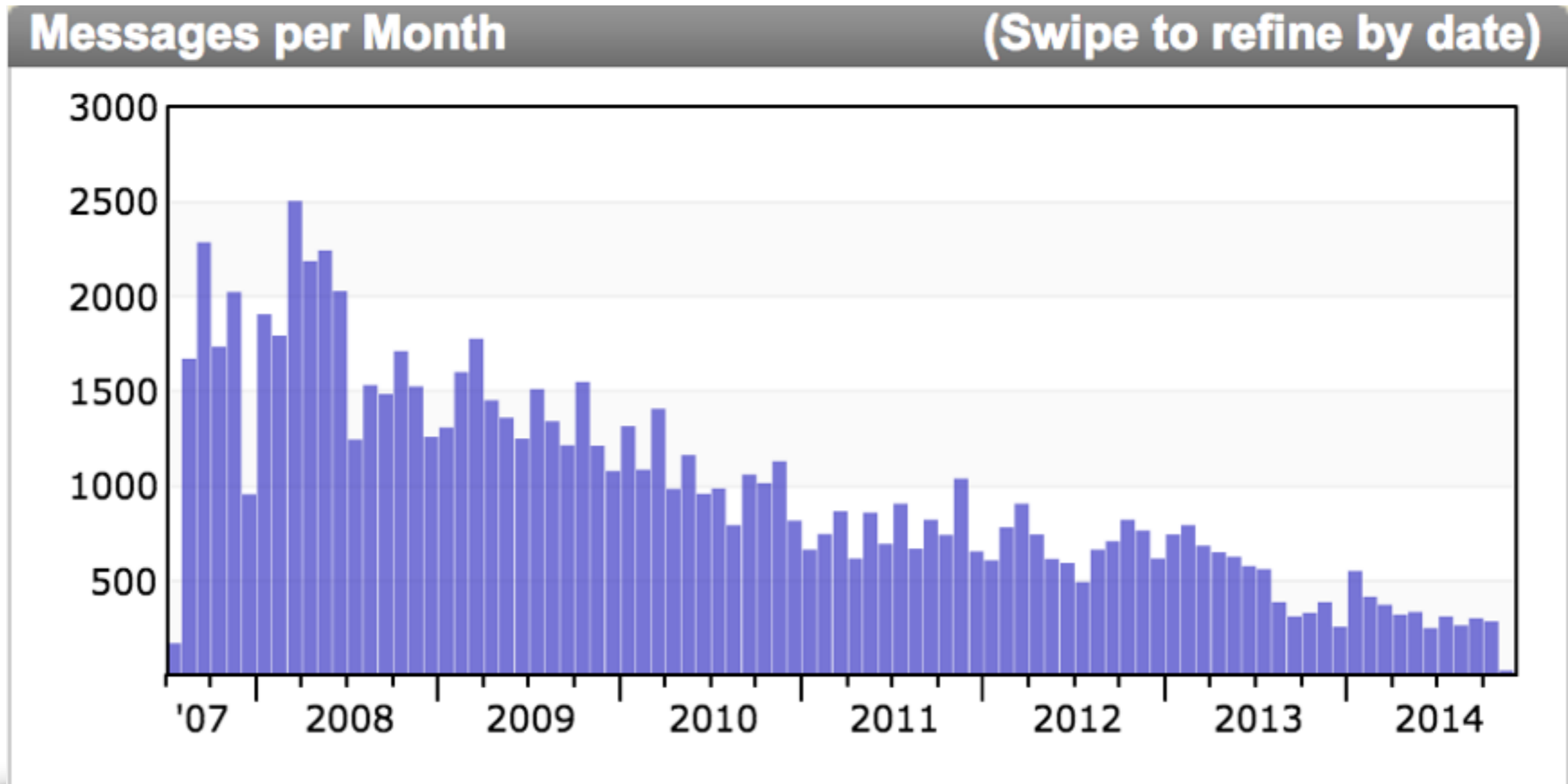
4. datetime

5. auth-roles

Downloads Over the Last 12 Months For org.apache.wicket

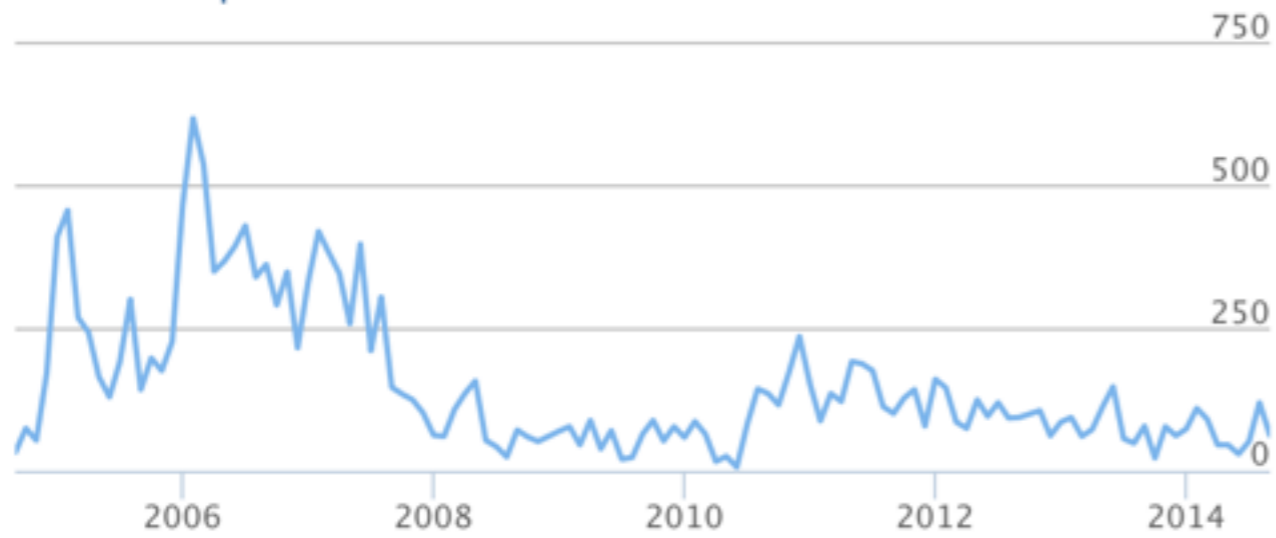


Mailinglist traffic

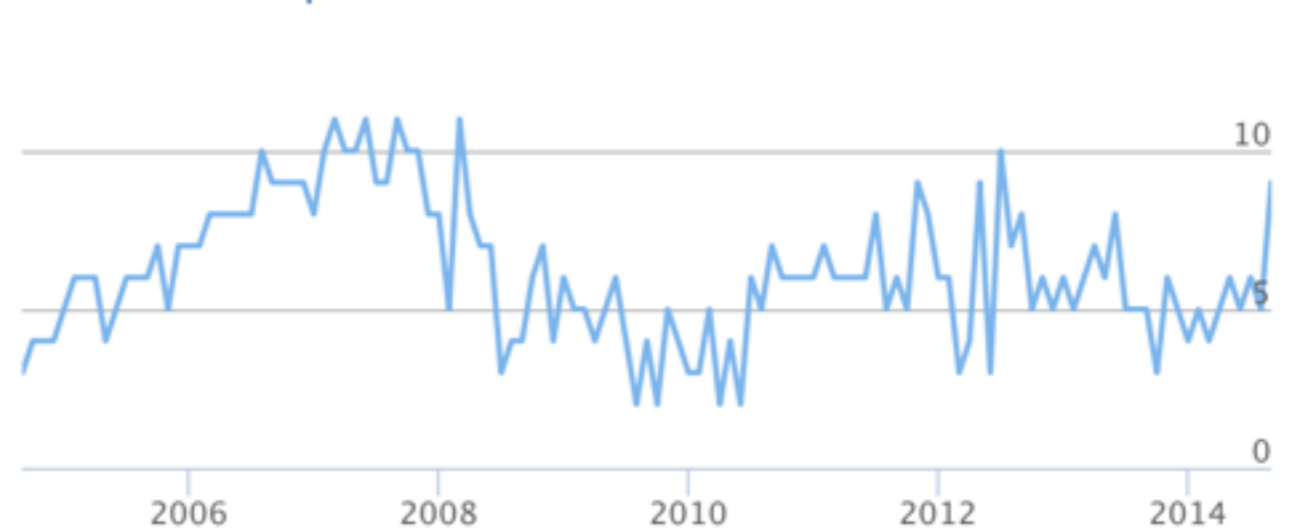


Commit activity

Commits per Month



Contributors per Month



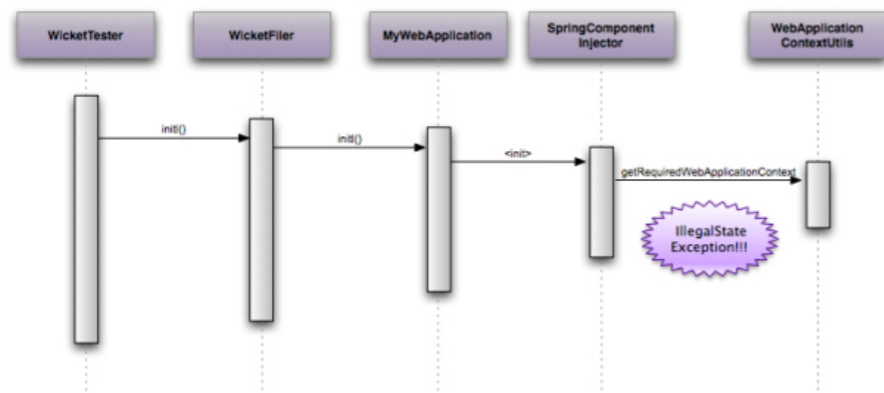
Meanwhile
at devoxx...

How do you create Webapps?

JSF/Java EE	###
REST + JavaScript	### ## ### ### ###
Play framework	###
Spring MVC	###
Widget	
JavaScript framework library	### ## ###
Grails	
GWT	
Vaadin	
APACHE SLING	
Websockets + JS	

User Manual

As you can see above, the Exception gets raised during the initialization of the `WicketTester` instance even before the actual test method gets executed. Even though we have applied rather cool and simple annotation based test configuration already described and passed in perfectly well prepared `ApplicationContext` instance to the `WicketTester` instance in the constructor, somewhere down the rabbit hole someone complained that no `WebApplicationContext` instance could have been found which seems to be required in order to initialize the `WicketTester` properly.



The problem that we run against here is due to the fact that `SpringComponentInjector` during its own initialization is trying to get hold of an according Spring's `ApplicationContext` instance that would normally be there in a runtime environment but does not find any since we are running in a test environment currently. `SpringComponentInjector` delegates to Spring's own `WebApplicationContextUtils` class to retrieve the instance of `ApplicationContext` out of the `ServletContext` which is perfectly fine for a runtime environment but is unfortunately failing in a test environment:

```
public static WebApplicationContext getRequiredWebApplicationContext(ServletContext sc)
    throws IllegalStateException {
    WebApplicationContext wac = getWebApplicationContext(sc);
    if (wac == null) {
        throw new IllegalStateException("No WebApplicationContext found: no
ContextLoaderListener registered?");
    }
    return wac;
}
```

If you still remember we defined a `ContextLoaderListener` in our `web.xml` file as part of the configuration of our runtime environment that makes sure an according `WebApplicationContext` instance gets initialized and registered against the `ServletContext` properly. Luckily, this problem can easily be solved if we slightly change the way we initialize `SpringComponentInjector` in our main `MyWebApplication` class. Apart from the constructor that we have used so far, there is another constructor in the `SpringComponentInjector` class that expects the caller to provide it with an according `ApplicationContext` instance rather than trying to resolve one on its own:

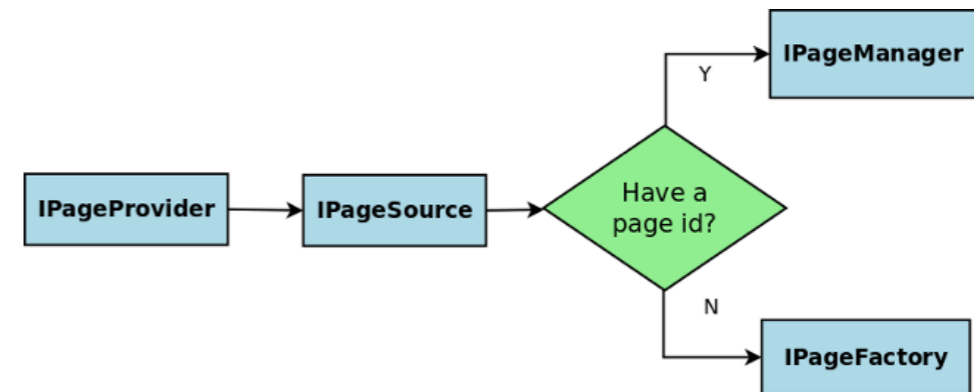
```
public SpringComponentInjector(WebApplication webapp, ApplicationContext ctx,
    boolean wrapInProxies)
{
    if (webapp == null)
    {
        throw new IllegalArgumentException("Argument [[webapp]] cannot be null");
    }
    if (ctx == null)
    {
        throw new IllegalArgumentException("Argument [[ctx]] cannot be null");
    }
    // store context in application's metadata ...
    webapp.setMetadata(CONTEXT_KEY, new ApplicationContextHolder(ctx));
    // ... and create and register the annotation aware injector
    InjectorHolder.setInjector(new AnnotSpringInjector(new ContextLocator(),
```

25 Wicket Internals

25.1 Page storing

During request handling, Wicket manages page instances through interface `org.apache.wicket.request.handler.IPageProvider`. This interface creates a new page instance or loads a previously serialized page instance if we provide the corresponding page id. `IPageProvider` delegates page creation and retrieval to interface `org.apache.wicket.request.mapper.IPageSource`. When page class is provided `IPageSource` delegates page creation to interface `org.apache.wicket.IPageFactory`, while when page id is provided it uses interface `org.apache.wicket.page.IPageManager` to load the previously serialized page.

The following workflow diagram summarizes the mechanism seen so far:



IPageManager

`org.apache.wicket.page.IPageManager`'s task is to manage which pages have been used in a request and store their last state in the backing stores, namely `IPageStore`. The default implementation `org.apache.wicket.page.PageStoreManager` collects all stateful pages which have been used in the request cycle (more than one page can be used in a single request if for example `setResponsePage()` or `RestartResponseException` is used). At the end of the request all collected page instances are being stored in the first level cache - http session. They are stored in http session attribute named "wicket:persistentPageManagerData-APPLICATION_NAME" and passed to the underlying `IPageStore`. When the next http request comes `IPageProvider` will ask for page with specific id and `PageStoreManager` will look first in the http session and if no match is found then it will delegate to the `IPageStore`. At the end of the second request the http session based cache is being overwritten completely with the newly used page instances.

To setup another `IPageManager` implementation use `org.apache.wicket.Application.setPageManagerProvider(IPageManagerProvider)`. The custom `IPageManager` implementation may or may not use `IPageStore/IDataStore`.

IPageStore

`org.apache.wicket.pageStore.IPageStore`'s role is to mediate the storing and loading of pages done by the underlying `IDataStore`. The default implementation `org.apache.wicket.pageStore.DefaultPageStore` pre-processes the pages before passing them to

In a Nutshell, Wicket...

... has had **18,255** commits made by **62** contributors representing **318,037** lines of code

... is mostly written in **Java** with a **well-commented** source code

... has a well established, **mature codebase** maintained by a **large development team** with **stable Y-O-Y commits**

... took an estimated **84 years** of effort (COCOMO model) starting with its **first commit** in **September, 2004** ending with its most recent commit 1 days ago

– openhub.net report for Wicket

Inloggen op Mijn OV-chipkaart

Om gebruik te kunnen maken van deze dienst logt u in met uw gebruikersnaam en wachtwoord van Mijn OV-chipkaart.

Nog geen Mijn OV-chipkaartaccount? [U kunt hier eenvoudig een account registreren](#)

Gebruikersnaam *

Wachtwoord

Inloggen

[Gebruikersnaam vergeten?](#)

[Wachtwoord vergeten?](#)

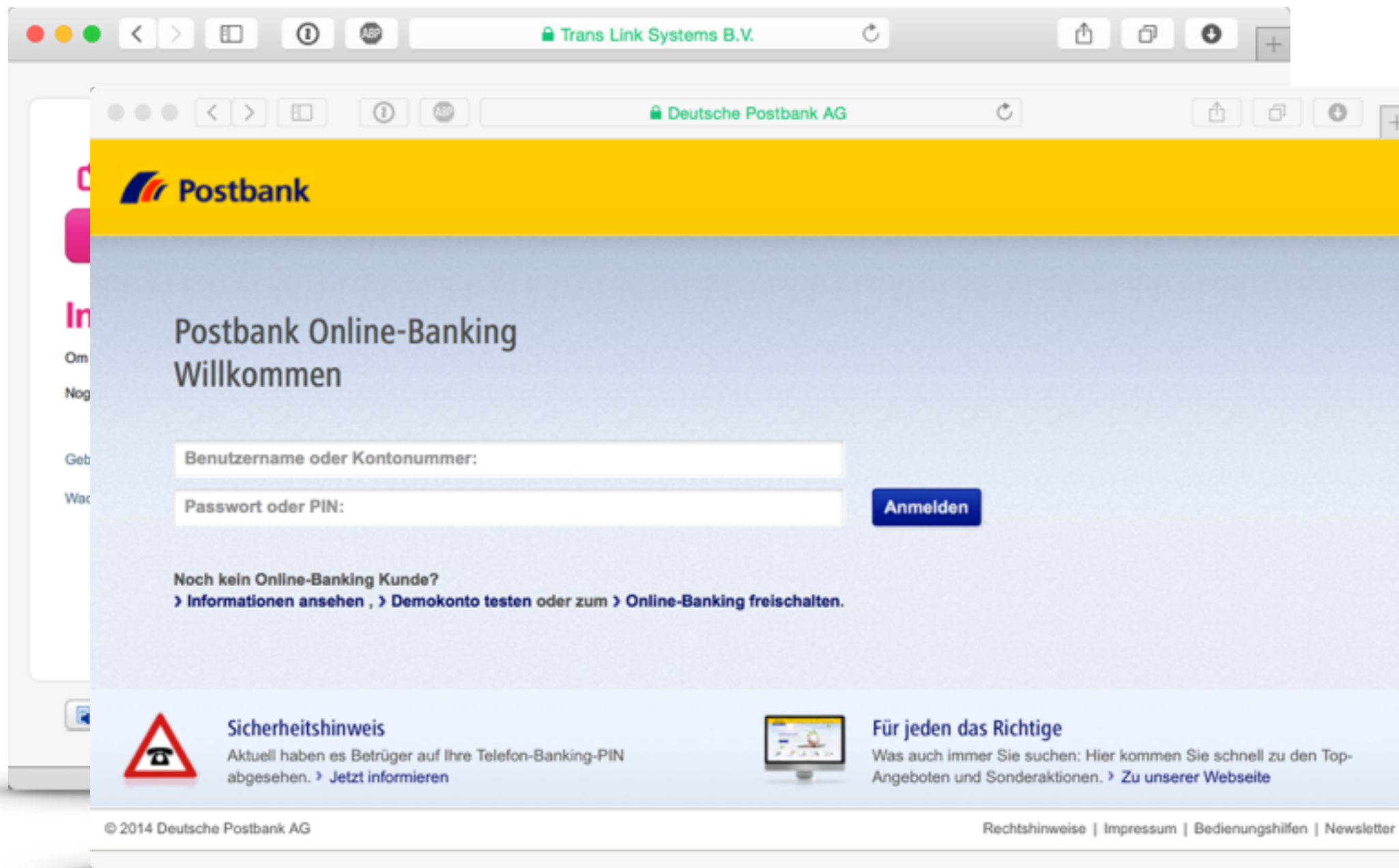
Contact

Vraag, klacht of tip over de OV-chipkaart?

- ▶ Vragen en antwoorden
- ▶ Stuur een e-mail
- ▶ Contact

Zie ook

- ▶ [Registreer account](#)
- ▶ [Inloggen lukt niet?](#)



Postbank Online-Banking Willkommen

Anmelden

Noch kein Online-Banking Kunde?
> [Informationen ansehen](#) , > [Demokonto testen](#) oder zum > [Online-Banking freischalten](#).



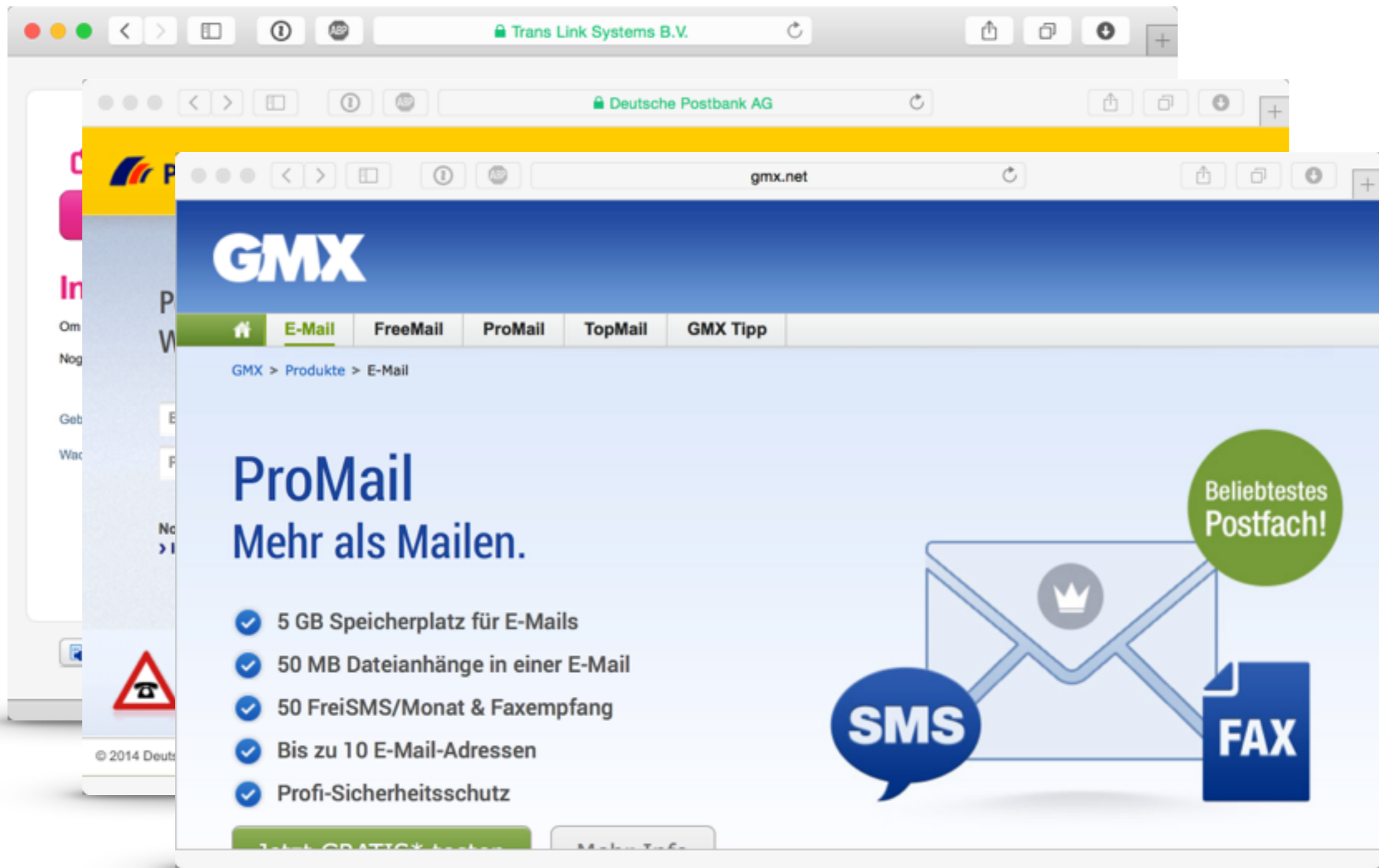
Sicherheitshinweis

Aktuell haben es Betrüger auf Ihre Telefon-Banking-PIN abgesehen. > [Jetzt informieren](#)



Für jeden das Richtige

Was auch immer Sie suchen: Hier kommen Sie schnell zu den Top-Angeboten und Sonderaktionen. > [Zu unserer Webseite](#)



GMX

Home E-Mail FreeMail ProMail TopMail GMX Tipp

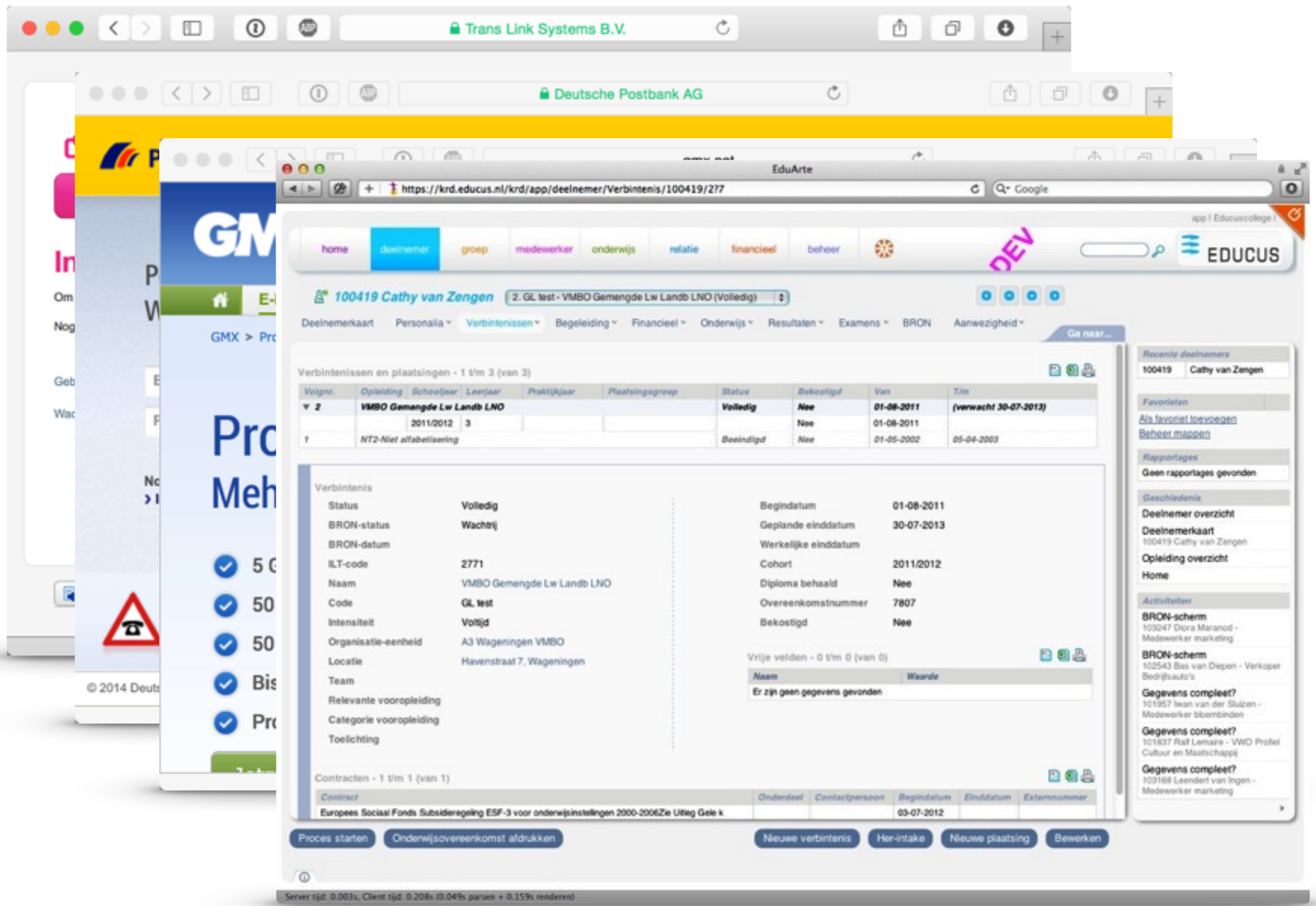
GMX > Produkte > E-Mail

ProMail Mehr als Mailen.

- ✓ 5 GB Speicherplatz für E-Mails
- ✓ 50 MB Dateianhänge in einer E-Mail
- ✓ 50 FreiSMS/Monat & Faxempfang
- ✓ Bis zu 10 E-Mail-Adressen
- ✓ Profi-Sicherheitsschutz



Jetzt GRATIS* testen Mehr Infos



home **deelnemer** groep medewerker onderwijs relatie financieel beheer

100419 Cathy van Zengen 2. GL test - VMBO Gemengde Lw Landb LNO (Volledig)

Deelnemerkaart Personalia **Verbindenissen** Begeleiding Financieel Onderwijs Resultaten Examens BRON Aanwezigheid

Verbindenissen en plaatsingen - 1 t/m 3 (van 3)

Volgnr.	Opleiding	Schooljaar	Leerjaar	Praktijkjaar	Plaatsingsgroep	Status	Bekostigd	Van	T/m
2	VMBO Gemengde Lw Landb LNO	2011/2012	3			Volledig	Nee	01-08-2011	(verwacht 30-07-2013)
1	NT2-Niet alfabetisering					Beëindigd	Nee	01-05-2002	05-04-2003

Verbindenis

Status	Volledig	Begindatum	01-08-2011
BRON-status	Wachtrij	Geplande einddatum	30-07-2013
BRON-datum		Werkelijke einddatum	
ILT-code	2771	Cohort	2011/2012
Naam	VMBO Gemengde Lw Landb LNO	Diploma behaald	Nee
Code	GL test	Overeenkomstnummer	7807
Intensiteit	Volledig	Bekostigd	Nee
Organisatie-eenheid	A3 Wageningen VMBO		
Locatie	Havenstraat 7, Wageningen		
Team			
Relevante vooropleiding			
Categorie vooropleiding			
Toelichting			

Vrije velden - 0 t/m 0 (van 0)

Naam	Waarde
Er zijn geen gegevens gevonden	

Contracten - 1 t/m 1 (van 1)

Contract	Onderdeel	Contactpersoon	Begindatum	Einddatum	Externnummer
Europees Sociaal Fonds Subsidie-regeling ESF-3 voor onderwijsinstellingen 2000-2006Zie Uitg Gele k			03-07-2012		

Proces starten Onderwijsovereenkomst afdrukken Nieuwe verbindenis Her-intake Nieuwe plaatsing Bewerken

Server tijd: 0.003s, Client tijd: 0.208s (0.049s parsen + 0.159s renderen)

The future of Wicket



Wicket 1.4

security releases only

Wicket 1.5

security releases only

Wicket 6

java 6

semantic
versioning

semi-monthly releases

mainly bug fixes

some new minor features

Wicket 7

Java 7

Servlet 3

Minor API breaks

Component queuing


```
public class HomePage extends WebPage {  
    public HomePage() {  
        add(new Label("title", "Hello!"));  
    }  
}
```

```
public class HomePage extends WebPage {  
    public HomePage() {  
        add(new Label("title", "Hello!"));  
    }  
}
```

```
<html>  
<body>  
    <h1 wicket:id="title"></h1>  
</body>  
</html>
```

```
public class HomePage extends WebPage {  
    public HomePage() {  
        add(new Label("title", "Hello!"));  
    }  
}
```

```
<html>  
<body>  
    <h1 wicket:id="title"></h1>  
</body>  
</html>
```

A screenshot of a web browser window. The address bar shows 'localhost'. The main content area displays the text 'Hello!' in a large, bold, black serif font.

Hello!

Add a `<div>` around content...


```
<html>
<body>
  <div wicket:id="container">
    <h1 wicket:id="title"></h1>
  </div>
</body>
</html>
```



```
<html>
```

```
<body>
```

```
  <div wicket:id="container">
```

```
    <h1 wicket:id="title"></h1>
```

```
  </div>
```

```
</body>
```

```
</html>
```

```
public class HomePage extends WebPage {  
  public HomePage() {  
    add(new WebMarkupContainer("container"))  
    add(new Label("title", "Hello!"));  
  }  
}
```

```
<html>
<body>
  <div wicket:id="container">
    <h1 wicket:id="title"></h1>
  </div>
</body>
</html>
```

```
pub
  }
}
```

Unexpected RuntimeException

Last cause: Unable to find component with id 'title' in [WebMarkupContainer]
Expected: 'container:title'.
Found with similar names: 'title'

Markup

The problem is in

["file:/Users/dashorst/Workspaces/luna/queuing/target/classes/com/mycompar"](file:/Users/dashorst/Workspaces/luna/queuing/target/classes/com/mycompar)

```
<div wicket:id="container">
  <h1 wicket:id="title"></h1>
</div>
```

```
bPage {
```

```
  container"))
```

```
  (!""));
```

in Wicket

code **always** follows markup

```
<html>  
<body>  
  <div wicket:id="container">  
    <h1 wicket:id="title"></h1>  
  </div>  
</body>  
</html>
```



```
<html>
<body>
  <div wicket:id="container">
    <h1 wicket:id="title"></h1>
  </div>
</body>
</html>
```

```
public class HomePage extends WebPage {
  public HomePage() {
    WebMarkupContainer container =
      new WebMarkupContainer("container");
    add(container);
    add(new Label("title", "Hello!"));
  }
}
```



```
<html>
<body>
  <div wicket:id="container">
    <h1 wicket:id="title"></h1>
  </div>
</body>
</html>
```

```
public class HomePage extends WebPage {
  public HomePage() {
    WebMarkupContainer container =
      new WebMarkupContainer("container");
    add(container);
    add(new Label("title", "Hello!"));
  }
}
```

```
<html>
<body>
  <div wicket:id="container">
    <h1 wicket:id="title"></h1>
  </div>
</body>
</html>
```

```
public class HomePage extends WebPage {
  public HomePage() {
    WebMarkupContainer container =
      new WebMarkupContainer("container");
    add(container);
    container.add(new Label("title", "Hello!"));
  }
}
```

in Wicket

code **always** follows markup

or does it?

Enter Component Queuing

(coming to you in wicket 7)

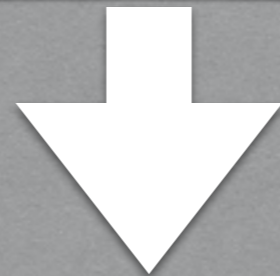
What if we delay adding components?

Queue components until their location in the markup hierarchy is known

```
markupContainer.add(new Label(...));  
markupContainer.queue(new Label(...));
```



```
public class HomePage extends WebPage {
    public HomePage() {
        WebMarkupContainer container =
            new WebMarkupContainer("container");
        add(container);
        container.add(new Label("title", "Hello!"));
    }
}
```



```
public class HomePage extends WebPage {
    public HomePage() {
        queue(new WebMarkupContainer("container"));
        queue(new Label("title", "Hello!"));
    }
}
```

```
<html>
<body>
  <div wicket:id="container">
    <h1 wicket:id="title"></h1>
  </div>
</body>
</html>
```

```
<html>
<body>
  <header><h1 wicket:id="title"></h1></header>
  <div wicket:id="container">
    ...
  </div>
</body>
</html>
```

Limitations of Component Queuing

```
queue(new Label("label", "Some text"));  
queue(new Label("label", "Other text"));
```



```
queue(new Label("label", "Some text"));
queue(new Label("label", "Other text"));
```

Unexpected RuntimeException

Last cause: A component with id: title has already been queued
WicketMessage: Can't instantiate page using constructor 'public com.mycompany.HomePage(org.apache.wicket...

Stacktrace

Root cause:

```
org.apache.wicket.WicketRuntimeException: A component with id: title has already been queued
  at org.apache.wicket.ComponentQueue.add(ComponentQueue.java:60)
  at org.apache.wicket.ComponentQueue.add(ComponentQueue.java:43)
  at org.apache.wicket.MarkupContainer.queue(MarkupContainer.java:1988)
  at com.mycompany.HomePage.<init>(HomePage.java:18)
  at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
  at org.apache.wicket.session.DefaultPageFactory.newPage(DefaultPageFactory.java:171)
  at org.apache.wicket.session.DefaultPageFactory.newPage(DefaultPageFactory.java:76)
```


Restrictions of Queuing: Ancestors

```
queue(new Label("first"));  
queue(new Label("last"));
```

```
WebMarkupContainer secure=new  
WebMarkupContainer("secure") {  
    public void onConfigure() {  
        super.onConfigure();  
        setVisible(isViewingOwnProfile());  
    }  
};
```

```
queue(secure);  
secure.queue(new Label("creditCardNumber"));  
secure.queue(new Label("creditCardExpiry"));
```

```
<h3>
```

```
Welcome,
```

```
<span wicket:id="first"></span>
```

```
<span wicket:id="last"></span>!
```

```
</h3>
```

```
<fieldset wicket:id="secure">
```

```
<legend>Secure</legend>
```

```
<dt>Card expiry: <i wicket:id="creditCardExpiry">
```

```
<dt>Card number: <i wicket:id="creditCardNumber">
```

```
</fieldset>
```

```
<h3>
```

Welcome,

```
<span wicket:id="first"></span>
```

```
<span wicket:id="last"></span>!
```

```
</h3>
```

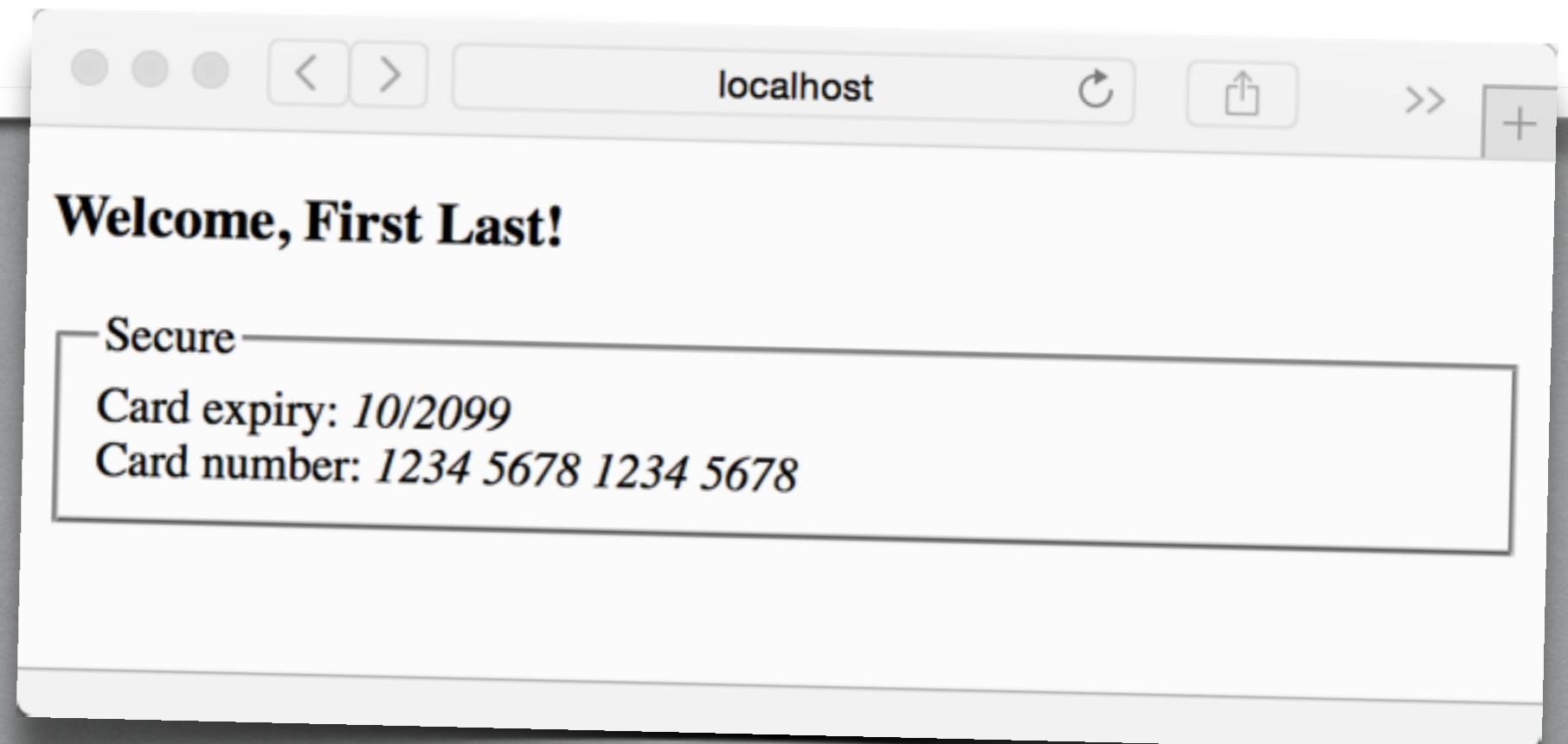
```
<fieldset wicket:id="secure">
```

```
<legend>Secure</legend>
```

```
<dt>Card expiry: <i wicket:id="creditCardExpiry">
```

```
<dt>Card number: <i wicket:id="creditCardNumber">
```

```
</fieldset>
```



```
<h3>
```

```
Welcome,
```

```
<span wicket:id="first"></span>
```

```
<span wicket:id="last"></span>!
```

```
</h3>
```

```
<fieldset wicket:id="secure">
```

```
<legend>Secure</legend>
```

```
<dt>Card expiry: <i wicket:id="creditCardExpiry">
```

```
</fieldset>
```

```
<dt>Card number: <i wicket:id="creditCardNumber">
```



```
<h3>
```

Welcome,

```
<span wicket:id="first"></span>
```

```
<span wicket:id="last"></span>!
```

```
</h3>
```

```
<fieldset wicket:id="secure">
```

```
<legend>Secure</legend>
```

```
<dt>Card expiry: <i wicket:id="creditCardExpiry">
```

```
</fieldset>
```

```
<dt>Card number: <i wicket:id="creditCardNumber">
```

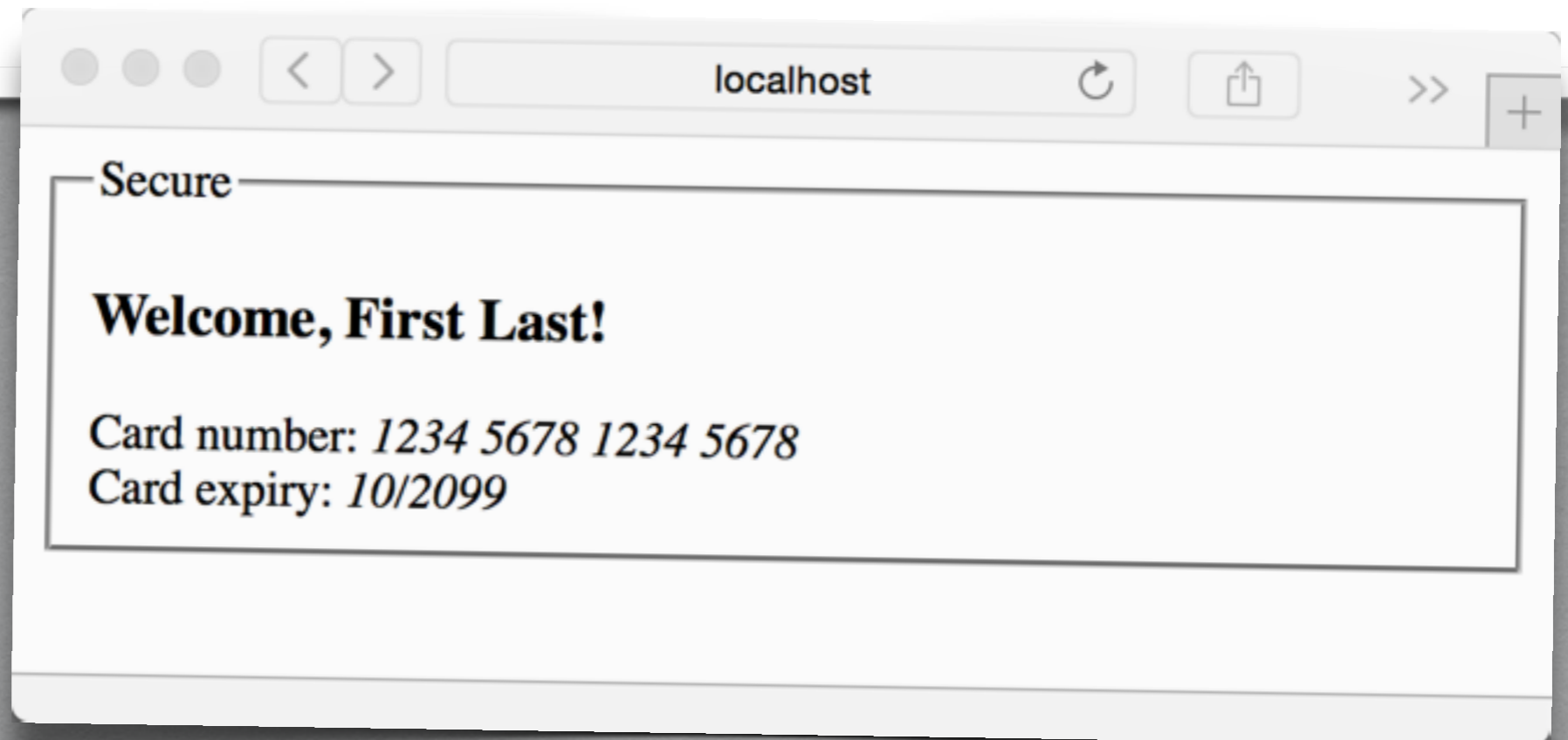
Unexpected RuntimeException

```
Last cause: Unable to find component with id 'creditCardNumber'  
Expected: ':creditCardNumber'.  
Found with similar names: "
```



```
<fieldset wicket:id="secure">
  <h3>
    Welcome,
    <span wicket:id="first"></span>
    <span wicket:id="last"></span>!
  </h3>
  <legend>Secure</legend>
  <dt>Card expiry: <i wicket:id="creditCardExpiry">
  <dt>Card number: <i wicket:id="creditCardNumber">
</fieldset>
```

```
<fieldset wicket:id="secure">
  <h3>
    Welcome,
    <span wicket:id="first"></span>
    <span wicket:id="last"></span>!
  </h3>
  <legend>Secure</legend>
  <dt>Card expiry: <i wicket:id="creditCardExpiry">
  <dt>Card number: <i wicket:id="creditCardNumber">
</fieldset>
```



Component Queuing

- Wicket 7 feature
- Prefer `queue()` when markup hierarchy can be altered
- Uniqueness of Wicket IDs still requirement at same level of queuing
- Can move down into child markup hierarchy
- Can't move up the markup hierarchy

Wicket 8

Java 8

PROJECT LAMBDA

“functional” programming in
Java

```
public interface ILinkListener {  
    void onLinkClicked();  
}
```

```
ILinkListener l = new ILinkListener(){  
    @Override  
    public void onLinkClicked() {  
        System.out.println("Klik");  
    }  
}
```

```
public interface ILinkListener {  
    void onLickClicked();  
}
```

```
ILinkListener l = () -> {  
    System.out.println("Klik");  
}
```

```
ILinkListener l = new ILinkListener(){  
    @Override  
    public void onLinkClicked() {  
        System.out.println("Klik");  
    }  
}
```

or

```
ILinkListener l = () -> {  
    System.out.println("Klik");  
}
```



```
add(new Link<Void>("save") {  
    @Override  
    public void onClick() {  
        dao.save(object);  
        getSession().info("Saved.");  
        setResponsePage(new OtherPage())  
    }  
});
```



```
add(new Link<Void>("save") {  
    @Override  
    public void onClick() {  
        dao.save(object);  
        getSession().info("Saved.");  
        setResponsePage(new OtherPage())  
    }  
});
```

no more type
information

```
add(new Link<>("save").onClick(()-> {  
    dao.save(object);  
    getSession().info("Saved.");  
    setResponsePage(new OtherPage())  
});
```

```
private void onSave() {  
    dao.save(object);  
    getSession().info("Saved.");  
    setResponsePage(new OtherPage())  
}
```



```
private void onSave() {  
    dao.save(object);  
    getSession().info("Saved.");  
    setResponsePage(new OtherPage())  
}
```

```
add(new Link<Void>("save")  
    .onClick(this::onSave));
```

A link with onclick, visibility and
body


```
add(new Link<Void>("like"));
```

```
add(new Link<Void>("like") {  
});
```

```
add(new Link<Void>("like") {  
    @Override  
    public void onClick() {  
        person.likedBy(me);  
    }  
});
```

```
add(new Link<Void>("like") {  
    @Override  
    public void onClick() {  
        person.likedBy(me);  
    }  
  
    @Override  
    public boolean isVisible() {  
        return person.isNotLiked();  
    }  
});
```

```
add(new Link<Void>("like") {  
    @Override  
    public void onClick() {  
        person.likedBy(me);  
    }  
  
    @Override  
    public boolean isVisible() {  
        return person.isNotLiked();  
    }  
}.setBody());
```



```
add(new Link<Void>("like") {  
    @Override  
    public void onClick() {  
        person.likedBy(me);  
    }  
  
    @Override  
    public boolean isVisible() {  
        return person.isNotLiked();  
    }  
}).setBody(new AbstractReadOnlyModel<String>() {  
}));
```

```
add(new Link<Void>("like") {
    @Override
    public void onClick() {
        person.likedBy(me);
    }

    @Override
    public boolean isVisible() {
        return person.isNotLiked();
    }
}).setBody(new AbstractReadOnlyModel<String>() {
    @Override
    public String getObject() {
    }
}));
```

```
add(new Link<Void>("like") {
    @Override
    public void onClick() {
        person.likedBy(me);
    }

    @Override
    public boolean isVisible() {
        return person.isNotLiked();
    }
}).setBody(new AbstractReadOnlyModel<String>() {
    @Override
    public String getObject() {
        StringBuilder sb = new StringBuilder("Like ");
        sb.append(person.getFirstName());
        return sb.toString();
    }
}));
```

A link with onclick, visibility and
body

```
add(new Link<>("like"));
```



```
add(new Link<>("like"))  
);
```

```
add(new Link<>("like")
    .visible(() -> person.isNotLiked())
);
```

```
add(new Link<>("like")
    .visible(() -> person.isNotLiked())
    .onClick(() -> person.likedBy(me))
);
```

```
add(new Link<>("like")
    .visible(() -> person.isNotLiked())
    .onClick(() -> person.likedBy(me))
    .body(() -> {
})
);
```

```
add(new Link<>("like")
    .visible(() -> person.isNotLiked())
    .onClick(() -> person.likedBy(me))
    .body(() -> {
        StringBuilder sb = new StringBuilder("Like ");
        sb.append(person.getFirstName());
        return sb.toString();
    })
);
```



```

add(new Link<Void>("like") {
    @Override
    public void onClick() {
        person.likedBy(me);
    }

    @Override
    public boolean isVisible() {
        return person.isNotLiked();
    }
}.setBody(new AbstractReadOnlyModel<String>() {
    @Override
    public String getObject() {
        StringBuilder sb = new StringBuilder("Like ");
        sb.append(person.getFirstName());
        return sb.toString();
    }
}));

```

```

add(new Link<>("like")
    .visible(() -> person.isNotLiked())
    .onClick(() -> person.likedBy(me))
    .body(() -> {
        StringBuilder sb = new StringBuilder("Like ");
        sb.append(person.getFirstName());
        return sb.toString();
    })
);

```

Anon inner classes: 17 lines

Java 8 lambdas: 9 lines

nashorn

JavaScript validation

```
ScriptEngineManager m  
    = new ScriptEngineManager();
```

```
ScriptEngineManager m  
    = new ScriptEngineManager();
```

```
ScriptEngine nashorn  
    = m.getEngineByName("nashorn");
```



```
ScriptEngineManager m
    = new ScriptEngineManager();

ScriptEngine nashorn
    = m.getEngineByName("nashorn");

nashorn.put("age", validatable.getValue());
```

```
ScriptEngineManager m
    = new ScriptEngineManager();

ScriptEngine nashorn
    = m.getEngineByName("nashorn");

nashorn.put("age", validatable.getValue());
String js = "age >= 18";
```

```
ScriptEngineManager m
    = new ScriptEngineManager();

ScriptEngine nashorn
    = m.getEngineByName("nashorn");

nashorn.put("age", validatable.getValue());
String js = "age >= 18";

Object result = nashorn.eval(js);
```

```
ScriptEngineManager m
    = new ScriptEngineManager();

ScriptEngine nashorn
    = m.getEngineByName("nashorn");

nashorn.put("age", validatable.getValue());
String js = "age >= 18";

Object result = nashorn.eval(js);
if(!((Boolean)result) {
    ValidationError e = new ValidationError();
    validatable.error(e);
}
```

```
ScriptEngineManager m
    = new ScriptEngineManager();

ScriptEngine nashorn
    = m.getEngineByName("nashorn");

nashorn.put("age", validatable.getValue());
String js = "age >= 18";

try {
    Object result = nashorn.eval(js);
    if (!(Boolean)result) {
        ValidationError e = new ValidationError();
        validatable.error(e);
    }
} catch (Exception e) {
}
```


java.time

Roadmap

Monthly releases

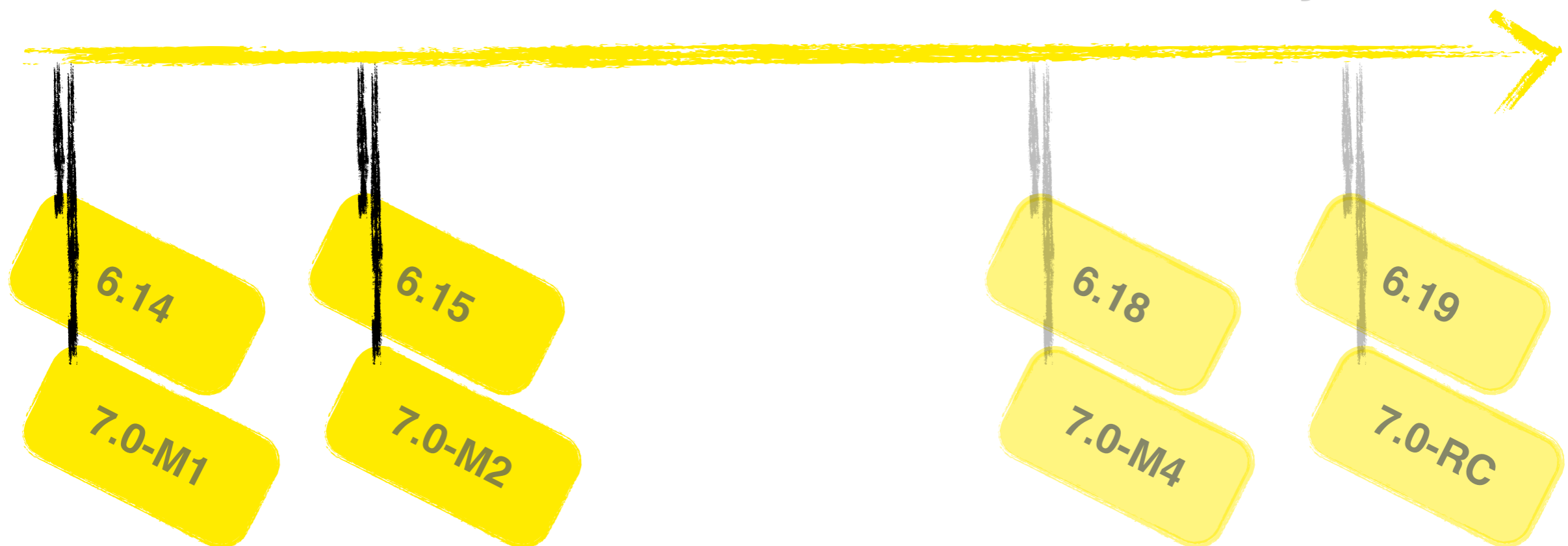
2014

feb

mar

nov

jan



2015

feb

mar

may

jun



Is there a future for
server-side frameworks?

TECHNOLOGY RADAR

[Radar home](#)
[Techniques](#)
[Tools](#)
[Platforms](#)
[Languages & frameworks](#)
[Radar A-Z](#)

● **ADOPT**

- 74. Clojure
- 75. Dropwizard
- 76. Scala, the good parts
- 77. Sinatra

● **TRIAL**

- 78. CoffeeScript
- 79. Go language **NEW**
- 80. Hive **NEW**
- 81. Play Framework 2
- 82. Reactive Extensions across languages **NEW**
- 83. Web API **NEW**

● **ASSESS**

- 84. Elixir **NEW**
- 85. Julia **NEW**
- 86. Nancy
- 87. OWIN
- 88. Pester **NEW**
- 89. Pointer Events **NEW**
- 90. Python 3 **NEW**
- 91. TypeScript **NEW**
- 92. Yeoman **NEW**

● **HOLD**

- 93. Handwritten CSS
- 94. JSF **NEW**



“We continue to see teams run into trouble using JSF -- JavaServer Faces -- and are recommending you avoid this technology.”

–ThoughtWorks Technology Radar January 2014

“We continue to see teams run into trouble using JSF -- JavaServer Faces -- and are recommending you avoid this technology. Teams seem to choose JSF because it is a J2EE standard without really evaluating whether the programming model suits them. We think JSF is flawed because it tries to abstract away HTML, CSS and HTTP, exactly the reverse of what modern web frameworks do. JSF, like ASP.NET webforms, attempts to create statefulness on top of the stateless protocol HTTP and ends up causing a whole host of problems involving shared server-side state. We are aware of the improvements in JSF 2.0, but think the model is fundamentally broken. We recommend teams use simple frameworks and embrace and understand web technologies including HTTP, HTML and CSS.”

–ThoughtWorks Technology Radar January 2014

“We continue to see teams run into trouble using JSF and are recommending you avoid this technology. We think JSF is flawed because it tries to abstract away HTML, CSS and HTTP. JSF, like ASP.NET webforms, attempts to create statefulness on top of the stateless protocol HTTP. We think the model is fundamentally broken.”

“We continue to see teams run into trouble using JSF and are recommending you avoid this technology. We think JSF is flawed because it tries to **abstract away** HTML, CSS and HTTP. JSF, like ASP.NET webforms, **attempts to create** statefulness on top of the stateless protocol HTTP. We think **the model** is fundamentally broken.”

“We recommend teams **use simple frameworks** and embrace and understand web technologies including HTTP, HTML and CSS.”

JavaScript Frameworks War

JQuery AngularJS Ember.js React

Backbone.js Meteor KnockoutJS Dojo

Agility.js CanJS Polymer Mithril

Ampersand Maria Flight Knockback.js

What is the Future for web
development?

What is the Future for web
development?

Web Components

What is the Future for web development?

Web Components

AngularJS
(2.0)

Polymer

What is the Future of Wicket?

Summary



a brief history
of Wicket



the current state
of Wicket



the future
of Wicket

Thank you!



Questions?



contact

@dashorst

topicus stand

