

Mensch-Maschine-Interaktion 2

HCI and the Web

Prof. Dr. Andreas Butz, Dr. Julie Wagner

HCI and the Web

- Organizational stuff
- Usability: a quick reminder
- How we use the web (device side)
- How we use the web (human side)
- HTML+CSS: a quick reminder
- Responsive Web Design: some Basics
- Responsive Web Design: Example and Exercise

Introducing the MMI2 team



Prof. Butz



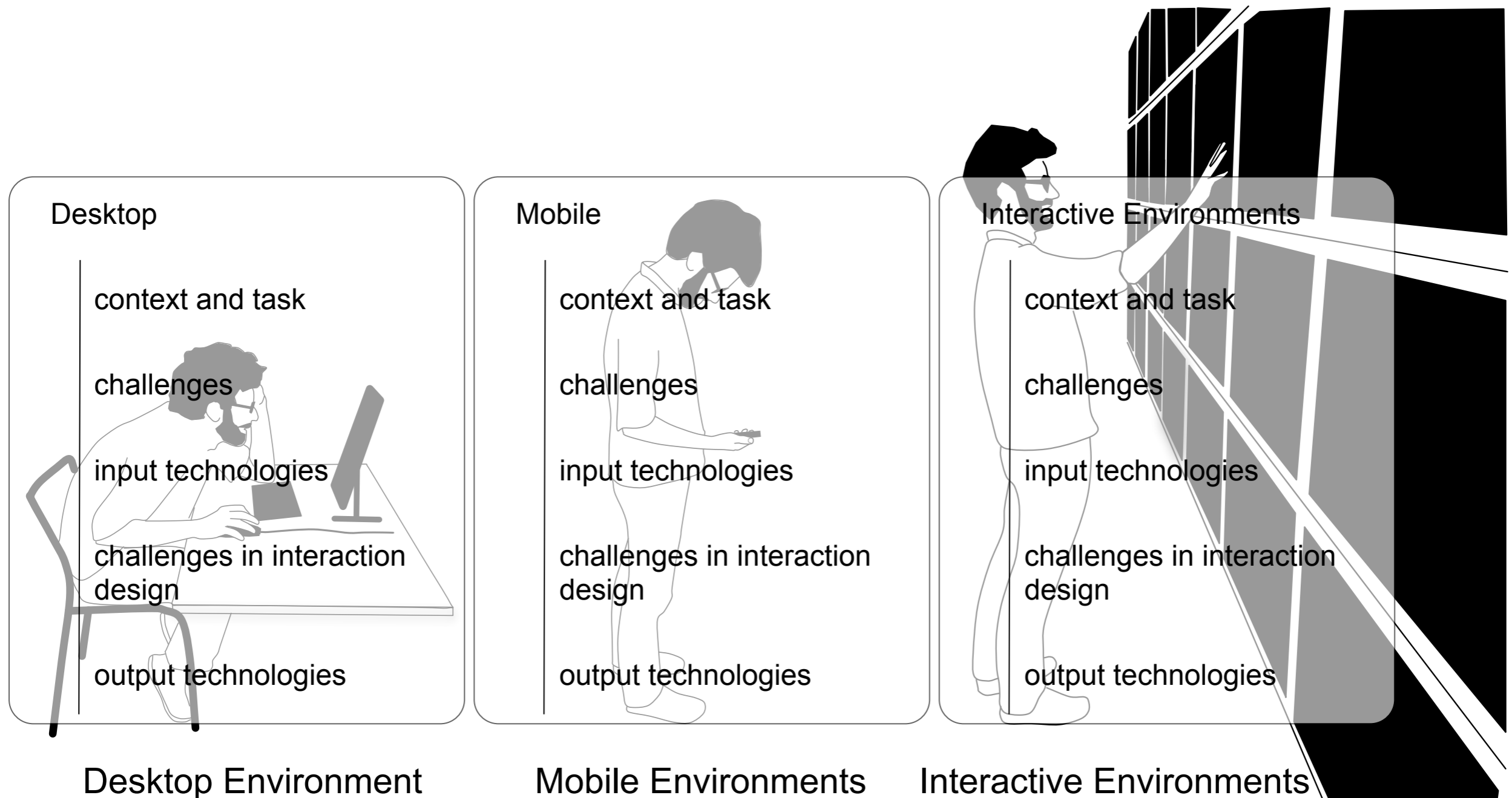
Dr. Julie Wagner



Sarah Thiel

Maraike Stuffer

Structure of Lecture



Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free!	
1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free!	
1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free!	
1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free! 1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free!	
1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free! 1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
5.2.14	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free!	
1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free! 1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Lecture

Datum	Thema der Vorlesung
16.10.13	Intro, Organization, Web
23.10.13	Guest Lecturer: Alexander Stage from Check 24
30.10.13	Desktop: context, tasks, challenges, input technologies
6.11.13	Desktop: challenges in interaction design, output technologies
13.11.13	Guest Lecturere:
20.11.13	Mobile Technologies:
27.11.13	Mobile Technologies:
4.12.13	Mobile Technologies:
11.12.13	Guest Lecturere:
18.12.13	Surprise session
25.12.13 - Free!	
1.1.14	
8.1.14	Interactive Environment
15.1.14	Interactive Environment
22.1.14	Interactive Environment
	Final Exam

Structure of Exercise

Please register for one of the exercises in UniWorx

■ Übung:

~~Montag 14-16 Uhr, Geschwister-Scholl-Platz 1, M-201~~

Montag 16-18 Uhr, Amalienstr. 73A, 101 - Tutor: Sarah Thiel

Dienstag 16-18 Uhr, Oettingenstr. 67, 067 - Tutor: Maraike Stuffer

Dienstag 18-20 Uhr, Oettingenstr. 67, 067 - Tutor: Maraike Stuffer

3 parts:

1. Summary of Lecture

2. Quiz (about summary of last week's exercise.)

3. Solution

- bring your equipment and program in class

- help each other

The screenshot shows a Doodle poll interface. At the top left is the 'Doodle' logo, and at the top right are links for 'CREATE USER-ACCOUNT' and 'SIGN IN'. Below the logo is the heading 'Mutually agree on a choice' and a sub-heading 'Enter your name in the input field below and select the options of your choice.' The main title of the poll is 'MMI2 Exercises: equipment'. Below the title, it says 'Poll initiated by Julie | 47 | 0 | 16 minutes ago' and 'Where: Munich'. The instruction reads 'Please indicate what programmable equipment you have.' Below this is a row of six blue buttons representing equipment options: 'Android phone or tablet', 'IOS phone or iPad + developer licence', 'Wii remote', 'Kinect', 'other PORTABLE device with accessible accelerometer data', and 'other PORTABLE device with accessible touch events'. In the bottom left corner, it indicates '47 participants'.

Rules for Plagiarism

- one trial leads to non-admission to the final exam
 - tools to check similarities of solutions
 - invitation to oral exam to determine if you are the true originator.
- think about your reputation if you plan to write master/ bachelor thesis at our chair.
- check for plagiarism can occur any time during the semester.

Exercise 1 (2 weeks time)

- design a responsive web design for the Olyphonics choir webpage (details later and in the exercise sheet)
- reward for best solution
 - Prof. Butz will call your phone to give his personal congratulations ;-)
 - Solution will be used on the web
 - Acknowledgement on the Olyphonics page

HCI and the Web

- Organizational stuff
- Usability: a quick reminder
- How we use the web (device side)
- How we use the web (human side)
- HTML+CSS: a quick reminder
- Responsive Web Design: some Basics
- Responsive Web Design: Example and Exercise

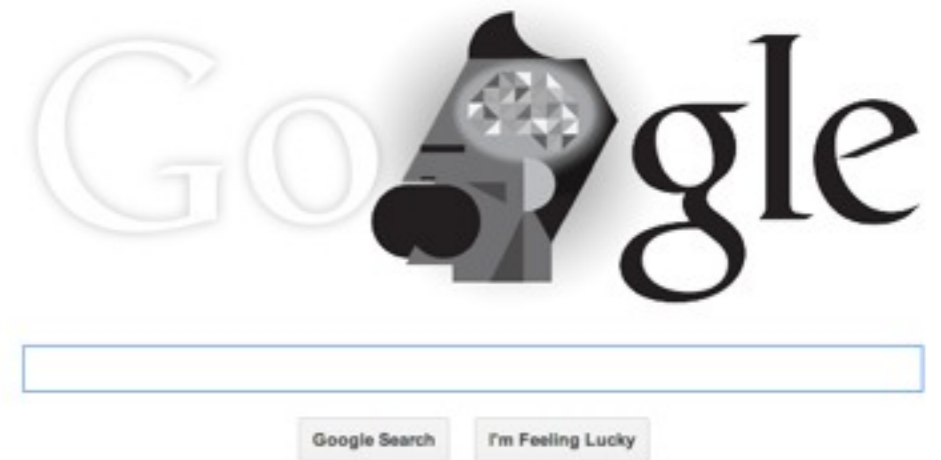
What is Usability?



- “Usability is a quality attribute that assesses how easy user interfaces are to use. The word ‘usability’ also refers to methods for improving ease-of-use during the design process.” (Jakob Nielsen)
- “Scientific discipline using observation, measurement and design principles to enhance a site visitor’s ability to perform specific tasks” (Kathy Gill)
- “... the effectiveness, efficiency and satisfaction with which a specified set of users can achieve a specified set of tasks ...” (ISO)

Why is Usability Important?

- Improving usability can
 - increase productivity of users
 - reduce costs (support, efficiency)
 - increase sales/revenue (web-shop)
 - enhance customer loyalty
 - win new customers

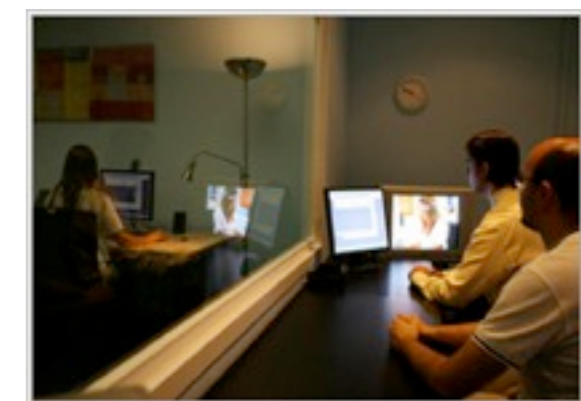
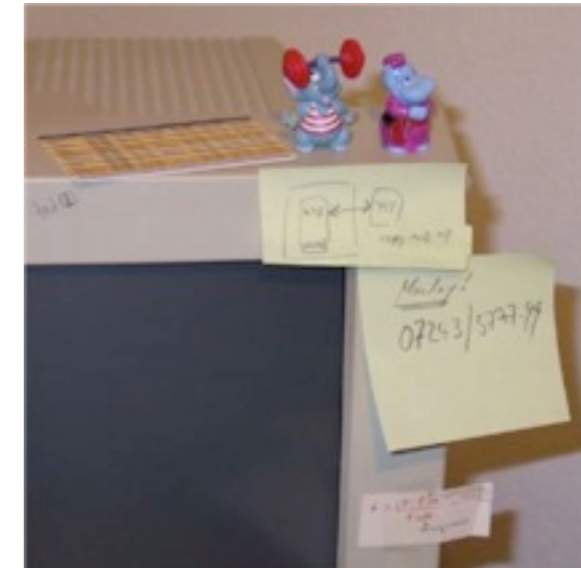


- Several case studies that show the benefit of usability
- Usability is often considered a sign of quality
- Working with users can create ideas for new products, e.g. "similarities" feature (people who bought this also bought that) at amazon.com (Source: Interview Maryam Mohit

http://www.medien.ifi.lmu.de/fileadmin/mimuc/mmi_ws0304/reading/2003-10-23_interview_mohit.pdf)

How do we achieve usability?

- Investigate requirements seriously
 - Observations, studies, focus groups
- Usability is a central element of all development activities
 - Part of quality assurance
- Iterative development
 - Early prototypes: Paper prototypes, mock-ups
 - High-fidelity prototypes & user studies
- Guidelines and principles
 - E.g. learnability, efficiency, memorability, errors, satisfaction (Nielsen)
- Evaluation
 - Usability engineering as an empirical discipline



Web Usability

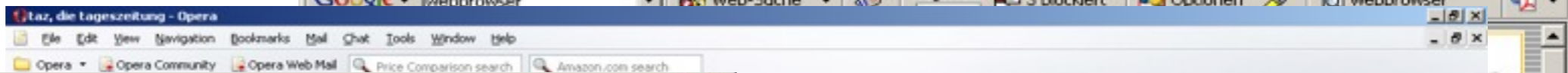
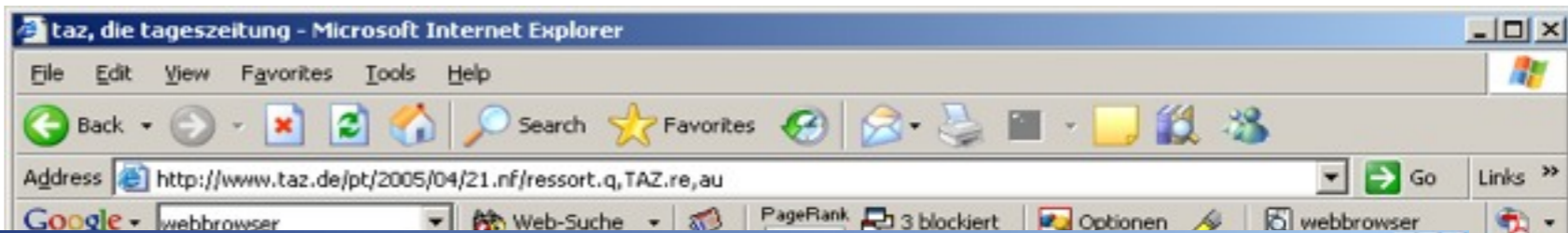
- Usability of Web sites and Web applications
 - delivered over the WWW
- Dependent on several issues related to
 - Web technology
 - Web design
 - Project Management
 - Usability evaluation
- Web usability is not about “adding some fancy graphics, color, and cool styles at the end of the project”
- Web usability can be measured!

HCI and the Web

- Organizational stuff
- Usability: a quick reminder
- How we use the web (device side)
- How we use the web (human side)
- HTML+CSS: a quick reminder
- Responsive Web Design: some Basics
- Responsive Web Design: Example and Exercise

The Web Means Heterogeneity of Platforms

- Processing power
 - Processor, co-processors, cache
 - RAM
- I/O-performance
 - Hard drive speed
 - Network
- Input and Output
 - Displays
 - Keyboard layout
- Additional Hardware and Periphery
 - Video and audio (in/out)
 - Card reader, printer, scanner
- Software
 - Browser
 - Operating System



taz, die tageszeitung - Mozilla Firefox

http://www.taz.de/pt/2005/04/21.nf/ressort.q,TAZ.re,au

Getting Started Latest Headlines

RUSSEN Komiker & Gurchy HITS 2 DISKO

drittwelt.de

taz-Abo digital --> PDF --> HTML --> TXT

FASTAFRICA

anzeigen tazshop taz-Genossenschaft über uns Kontakt

21.4.2005

Magazin | Le Monde diplomatique | TomTunnel

die tageszeitung 21.4.2005

| Startseite | Regionalausgaben | tazmog, Magazin | Le Monde diplomatique | TomTunnel |

Inhalt taz überregional

Seite 1

Aktuelles

Themen des Tages

Inland

Wirtschaft und Umwelt

Ausland

Meinung und Diskussion

Kultur

tazzwei

Leibesübungen

Ausland

Rücktritt und Neustart Berlusconi

Der italienische Ministerpräsident wirft aus taktischen Gründen das Handtuch, denn er will mit seinen alten Partnern eine neue Koalitionsregierung bilden. Doch die Parteien sind völlig zerstritten. Daher sind kurzfristige Neuwahlen nicht auszuschließen

115 Zeilen, MICHAEL BRAUN (TAZ-Bericht)

Neue Vorwürfe gegen Bolton

Google search 100%

taz-Abo digital --> PDF --> HTML --> TXT

FASTAFRICA

anzeigen tazshop taz-Genossenschaft über uns Kontakt

21.4.2005

Magazin | Le Monde diplomatique | TomTunnel

n das Handtuch, denn er will mit seinen alten Partnern eine neue ritten. Daher sind kurzfristige Neuwahlen nicht auszuschließen

die Abstimmung über den umstrittenen John Bolton als US-Botschafter

im umkämpften Nord-Darfur mit Schweizer Beteiligung

12 Seiten **Weltmusik** 30.4.

taz-Veranstaltungen

nrtaz retten abonnieren 1.000 Abos bis zum 30. Juni

Sind Sie eine Heldin?

extrablatt

Falls Sie sich tatsächlich fragen sollten, warum Sie eine Zeitung wie die taz brauchen, legen wir Ihrem Probeabo noch ein Buch mit den Antworten obendrauf. **5 Wochen taz plus Buch: 15 Euro**

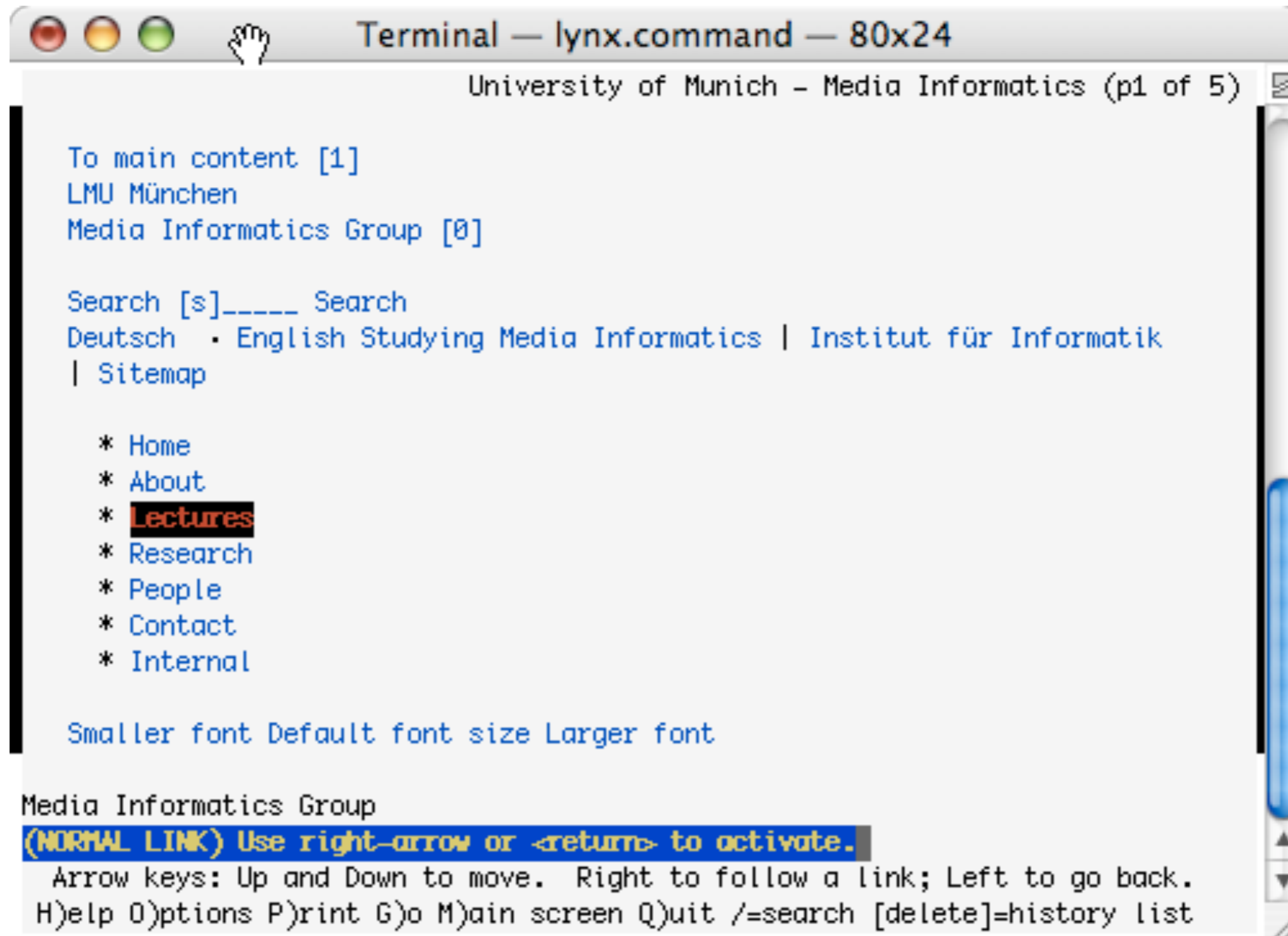
Other Graphical Browsers (2005 ;-)



Other Graphical Browsers (2013 version)



Plain Text Browser, e.g. Lynx



The image shows a terminal window titled "Terminal — lynx.command — 80x24". The content is a plain text rendering of a website. At the top, it says "University of Munich - Media Informatics (p1 of 5)". Below that, there are links for "To main content [1]", "LMU München", and "Media Informatics Group [0]". A search bar is present with the text "Search [s]_____ Search". There are also links for "Deutsch", "English Studying Media Informatics", "Institut für Informatik", and "Sitemap". A list of menu items follows: "* Home", "* About", "* Lectures" (highlighted in red), "* Research", "* People", "* Contact", and "* Internal". At the bottom, there are font size controls: "Smaller font Default font size Larger font". A status bar at the very bottom contains the text: "(NORMAL LINK) Use right-arrow or <return> to activate." followed by a list of keyboard shortcuts: "H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list".

```
Terminal — lynx.command — 80x24
University of Munich - Media Informatics (p1 of 5)

To main content [1]
LMU München
Media Informatics Group [0]

Search [s]_____ Search
Deutsch · English Studying Media Informatics | Institut für Informatik
| Sitemap

* Home
* About
* Lectures
* Research
* People
* Contact
* Internal

Smaller font Default font size Larger font

Media Informatics Group
(NORMAL LINK) Use right-arrow or <return> to activate.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

See also: <http://www.golem.de/news/wie-1992-cern-laesst-den-line-mode-browser-wiederauferstehen-1310-101914.html>

Audio Browsing

- There are users who listen to Web sites!
- Example:
 - Web browser Safari
 - Screen reader Voice Over (built into Mac OS)
- Who among the Web designers will think of these users?



VoiceOver

Your Mac speaks for itself.



HCI and the Web

- Organizational stuff
 - Usability: a quick reminder
 - How we use the web (device side)
 - How we use the web (human side)
-
- HTML+CSS: a quick reminder
 - Responsive Web Design: some Basics
 - Responsive Web Design: Example and Exercise

The Web Means Heterogeneity of Users

- In principle, anybody can use the Web!
- Huge span of user variety:
 - Kids
 - Beginners
 - Elderly
 - Experienced technically educated professionals
 - Technically ill-educated professionals
 - Hackers
- “Know Thy User” - Is it possible on the Web?
- Why do people use the Web?
 - Assumedly easy and simple way of achieving things
 - Because it is fun
 - Because there are no other options
- (As always:) Simplicity is most important

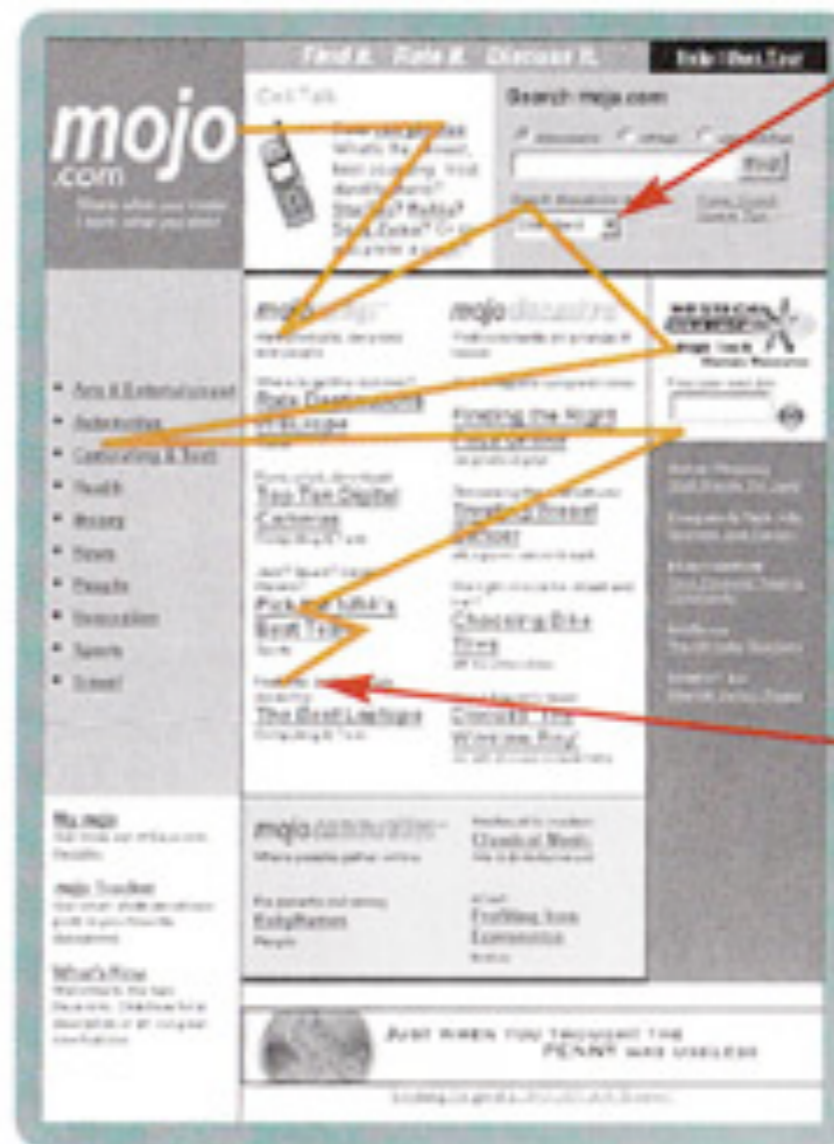
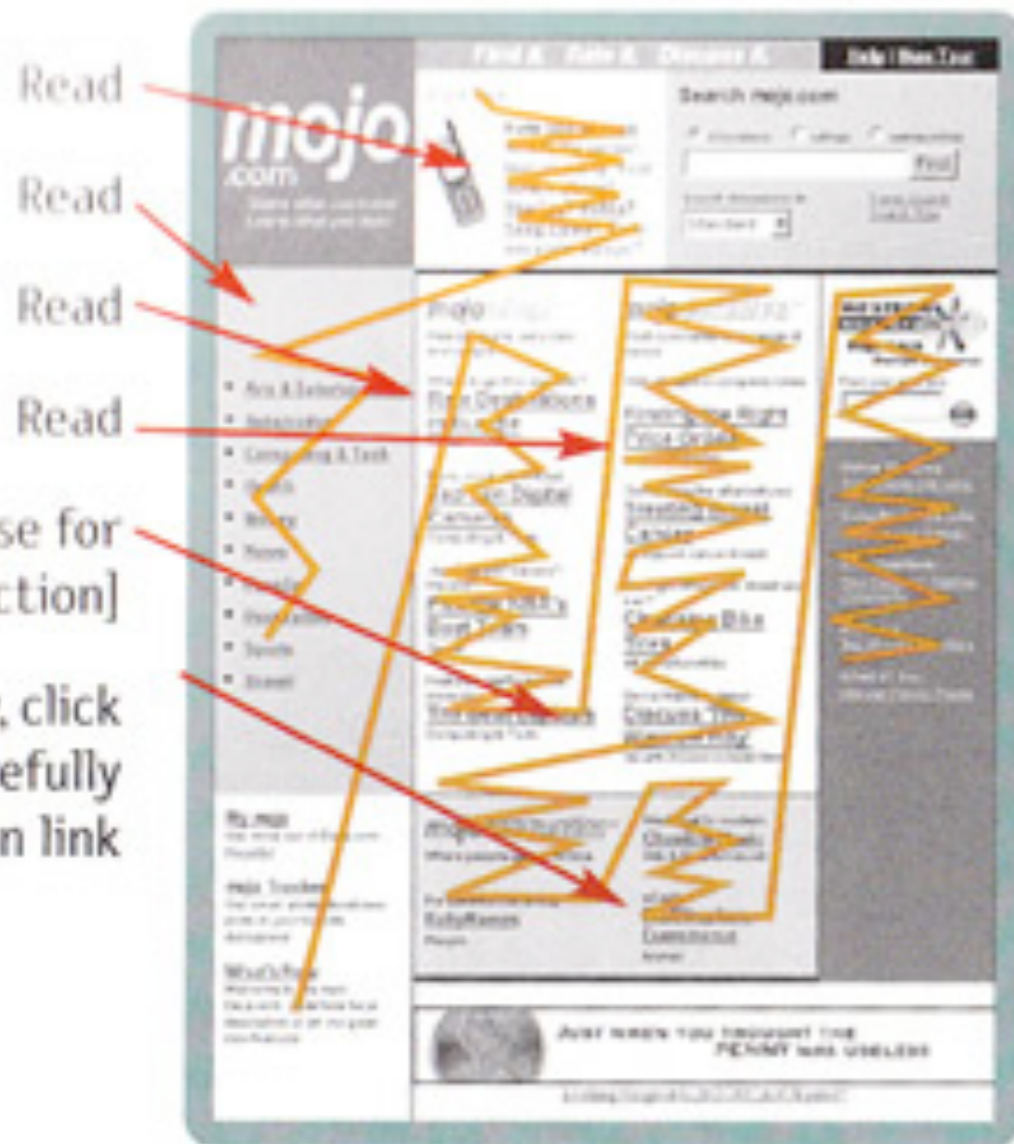


<http://www.friday-ad.co.uk/PhotoAds/LandingPages/image/child-pc.jpg>

Steve Krug: Design and Reality

WHAT WE DESIGN FOR...

THE REALITY...



Look around feverishly for anything that

a) is interesting, or vaguely resembles what you're looking for, and

b) is clickable.

As soon as you find a halfway-decent match, click.

If it doesn't pan out, click the Back button and try again.

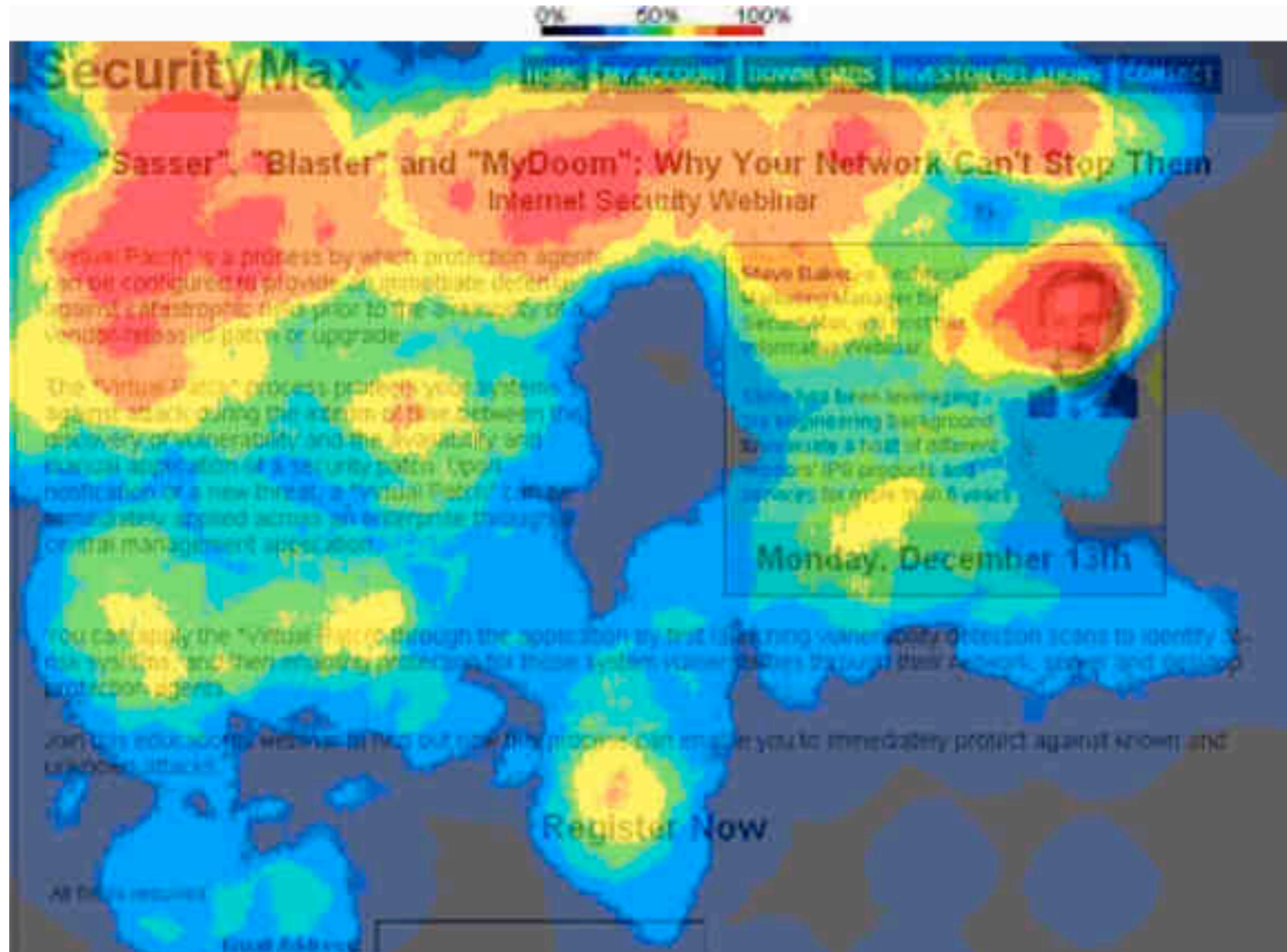
Don't make me think! Web Usability: Das intuitive Web, Steve Krug ISBN-10: 3826608909

Steve Krug: We Don't Read Pages, We Scan Them

- We are in a hurry.
- We know that we do not have to read everything.
- We are educated in scanning things.



Heatmaps to analyze web page usage



Steve Krug: We **Satisfice** (satisfying & sufficing)

- We do not make optimal choices
 - We are in a hurry.
 - There is not much penalty for guessing wrong.
 - Weighing options does not guarantee success.
 - Guessing is more fun.
- Gary Klein: Sources of Power - How People Make Decisions
 - Example: Fire commanders do rarely compare options!
 - Find a reasonable plan
 - Check it for obvious problems
 - Try it!
- see also: Daniel Kahneman: Thinking, Fast and Slow

Steve Krug: We Muddle Through

- Users in general do not care how and why things work
 - Any working solution is accepted
 - We do not have the time to analyze the details
 - There is no incentive for having it understood better
- Example:
 - Use a search box for navigating to a site



HCI and the Web

- Organizational stuff
 - Usability: a quick reminder
 - How we use the web (device side)
 - How we use the web (human side)
- HTML+CSS: a very (!!) quick reminder
 - Responsive Web Design: some Basics
 - Responsive Web Design: Example and Exercise

HTML Recap

- For details see lecture „Web-Informationssysteme“
 - <http://www.pms.ifi.lmu.de/lehre/webinfosys/13ws14/>
 - <http://videoonline.edu.lmu.de/en/node/1668>

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Title of the document</title>
  </head>

  <body>
    Content of the document.....
  </body>

</html>
```

http://www.w3schools.com/html/html5_intro.asp

CSS recap

- Main idea: Separate format from content!
 - content in HTML file
 - format in Cascading Style Sheet (CSS file)
- Style definitions for the tags in the HTML file
 - defaults if nothing is given
 - can be nested and become more and more specific
 - details again in the WebInfo lecture

```
body{font-size:75%;  
font-family:verdana,arial,'sans serif';  
background-color:#FFFFFF0;  
color:#000080;  
margin:10px;}
```

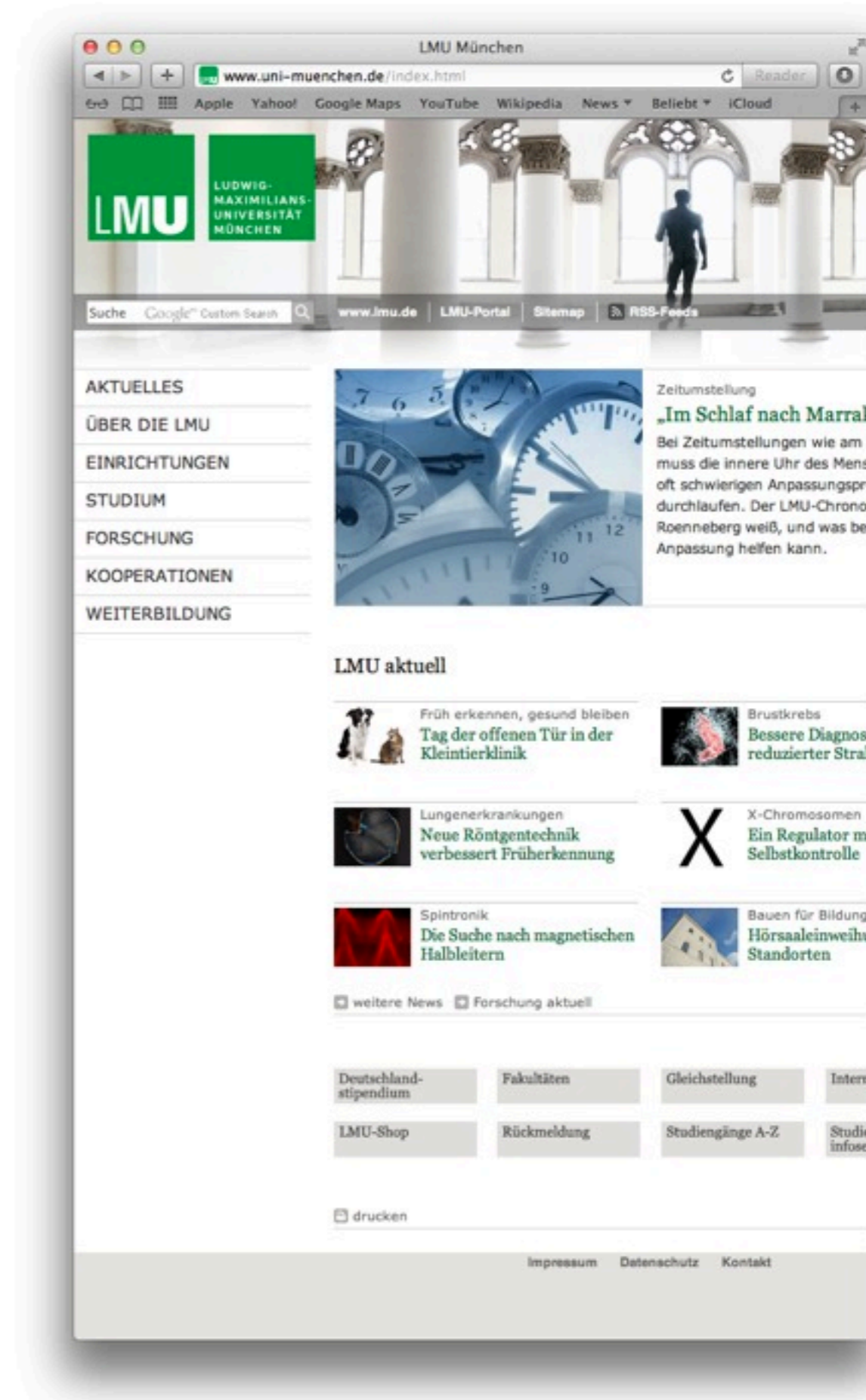
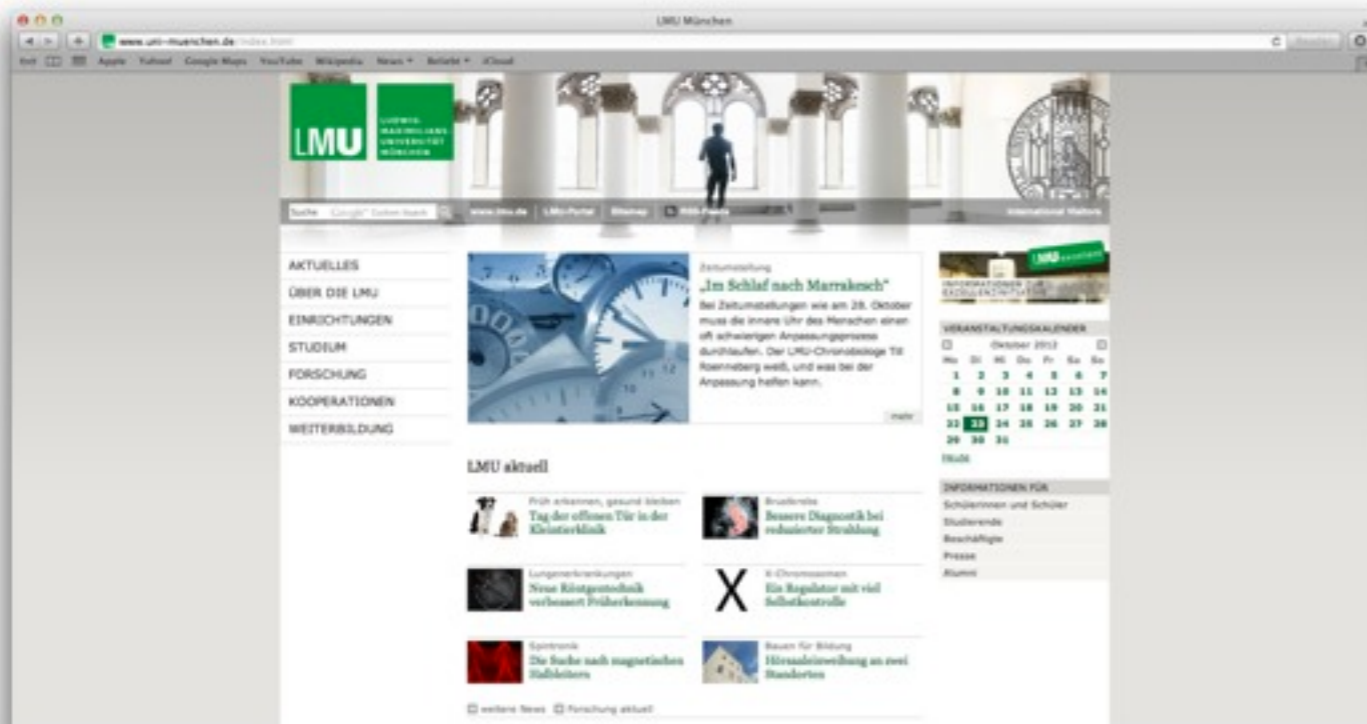
```
h1 {font-size:200%;}  
h2 {font-size:140%;}  
h3 {font-size:110%;}
```

http://www.w3schools.com/css/demo_default.htm

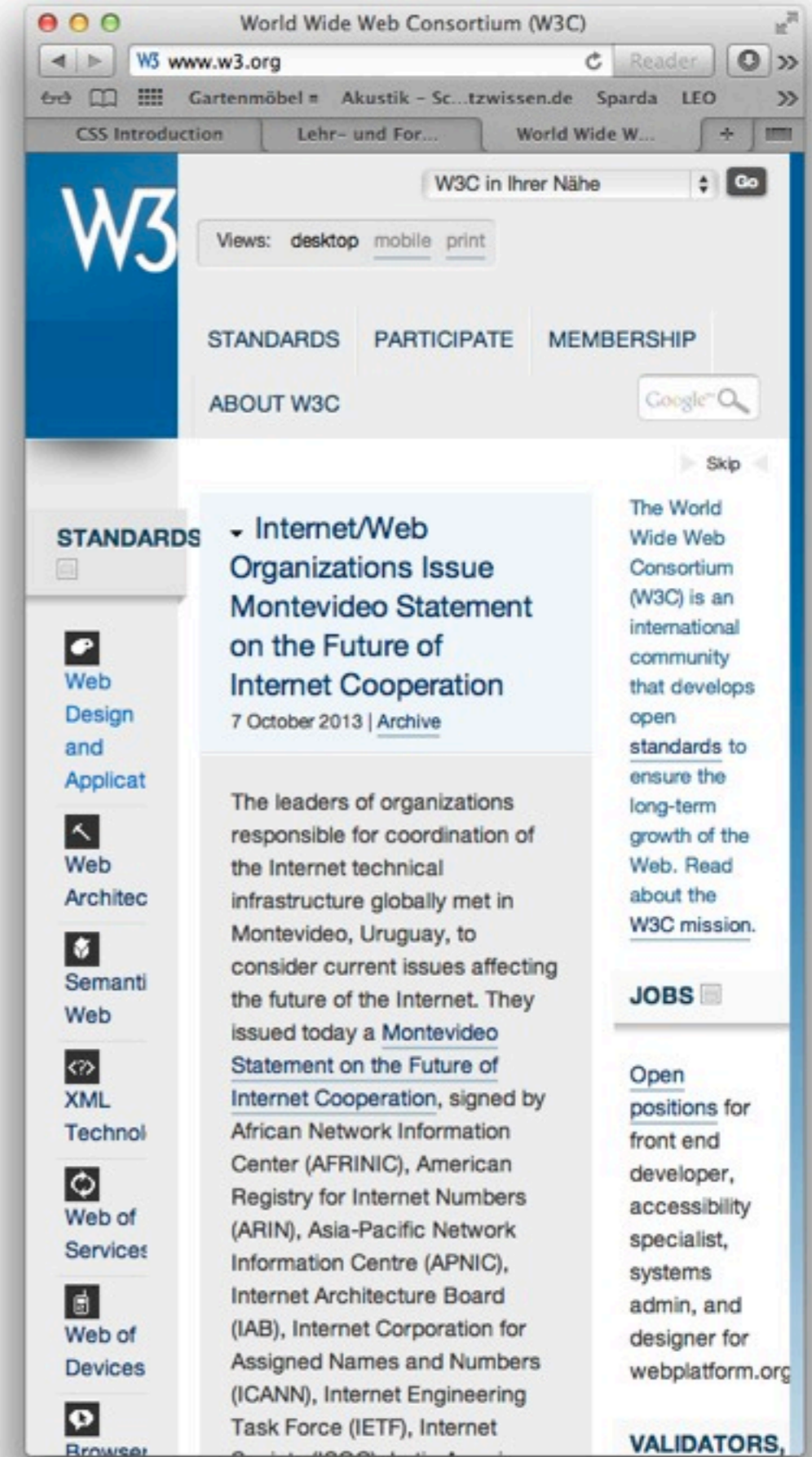
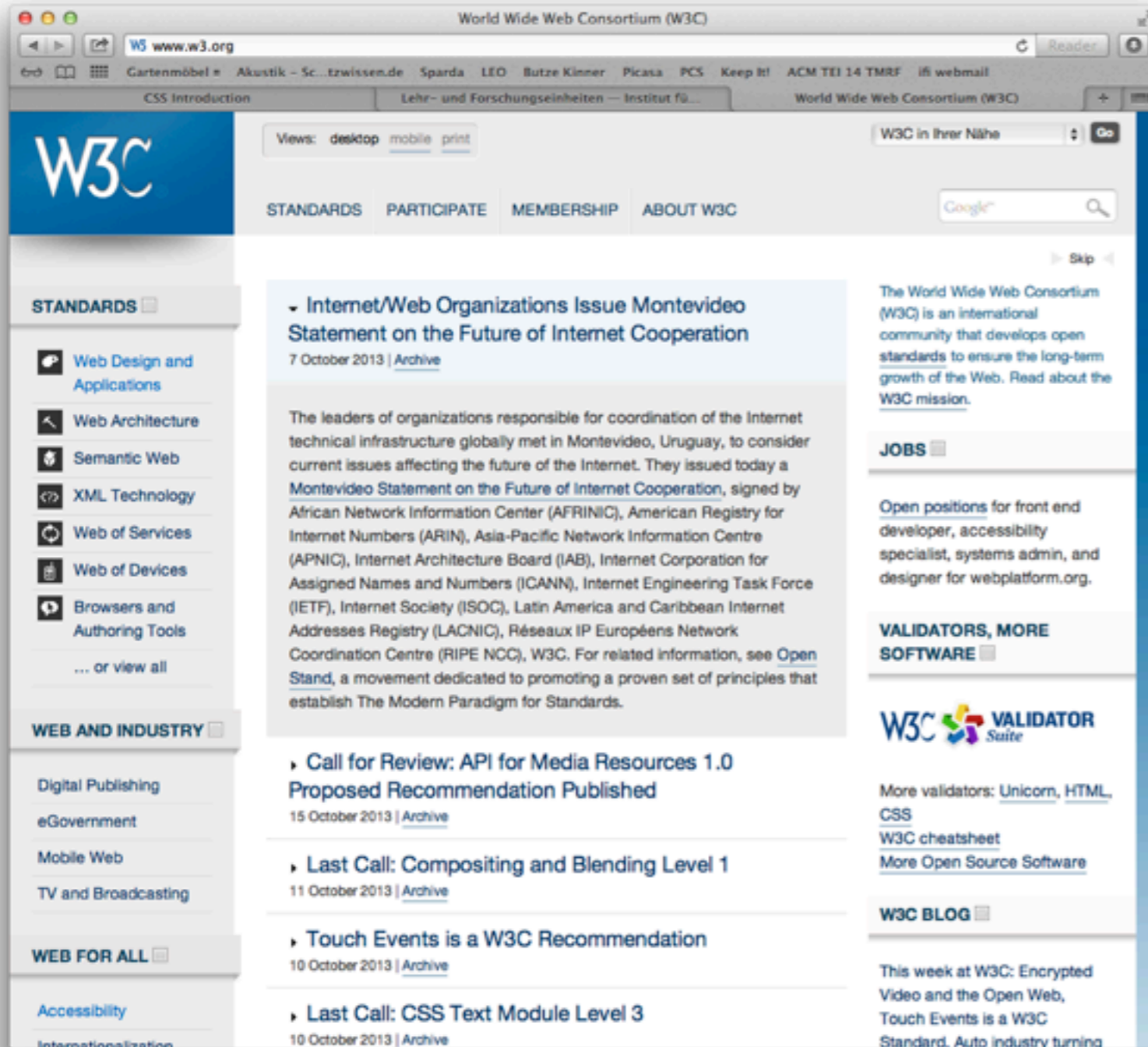
HCI and the Web

- Organizational stuff
- Usability: a quick reminder
- How we use the web (device side)
- How we use the web (human side)
- HTML+CSS: a quick reminder
- Responsive Web Design: some Basics
- Responsive Web Design: Example and Exercise

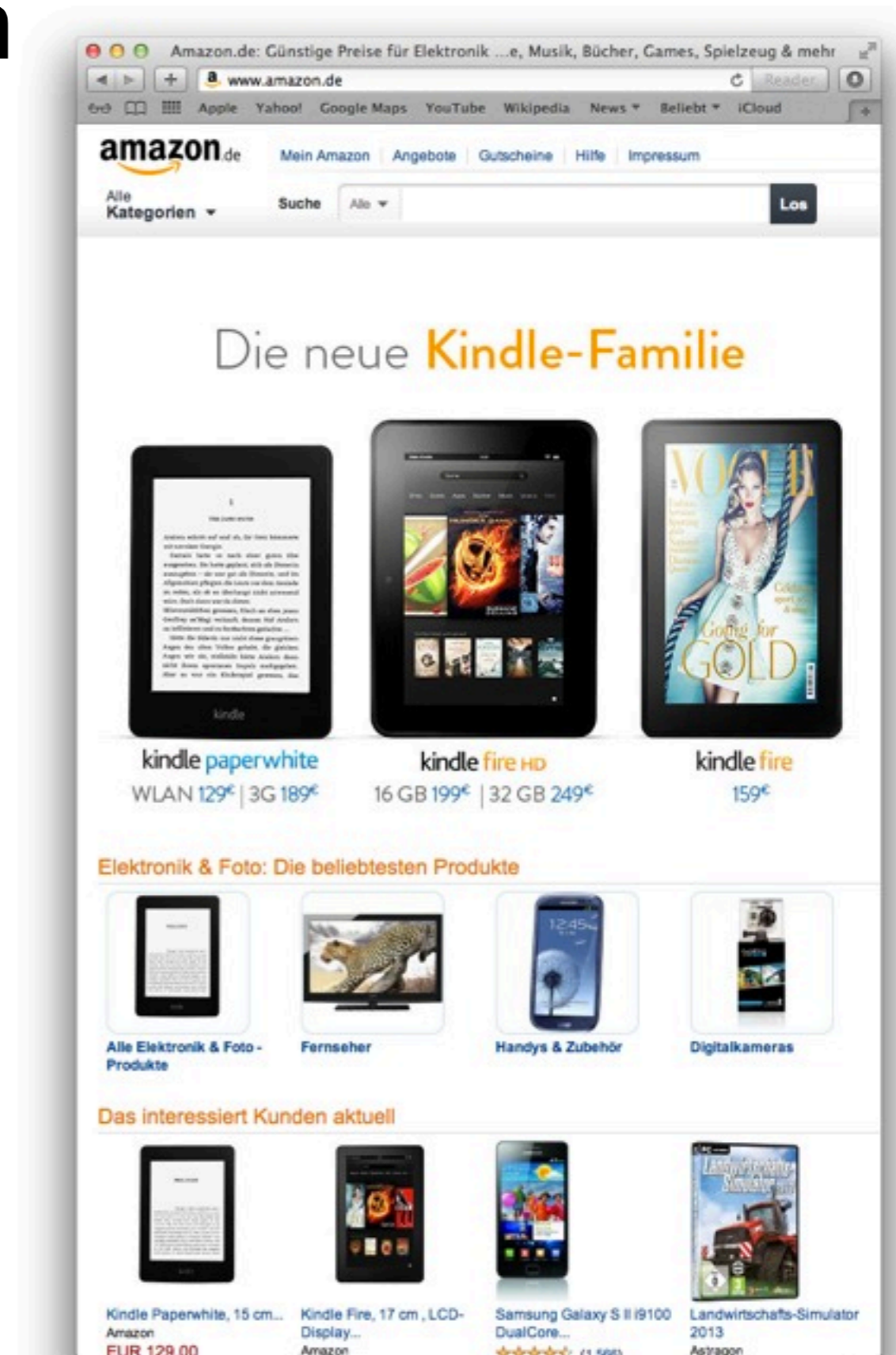
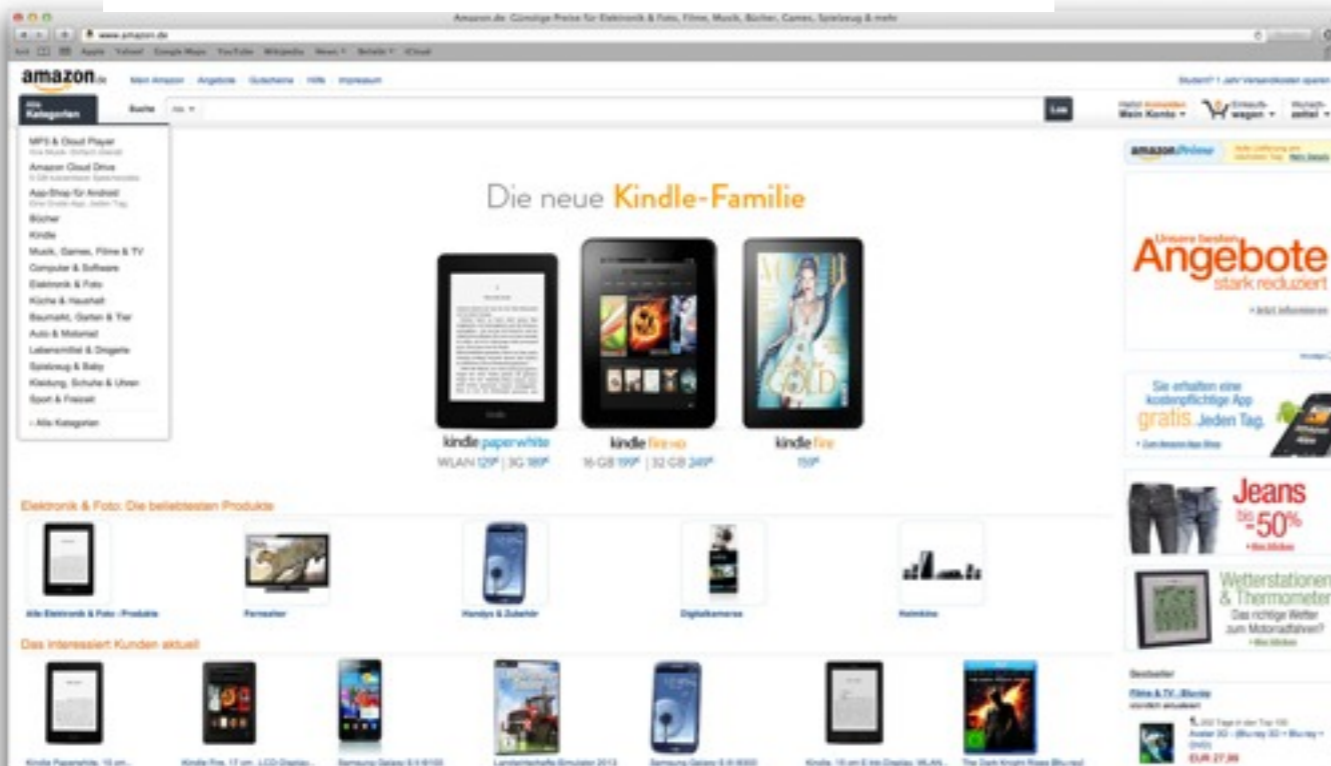
Static Web Design



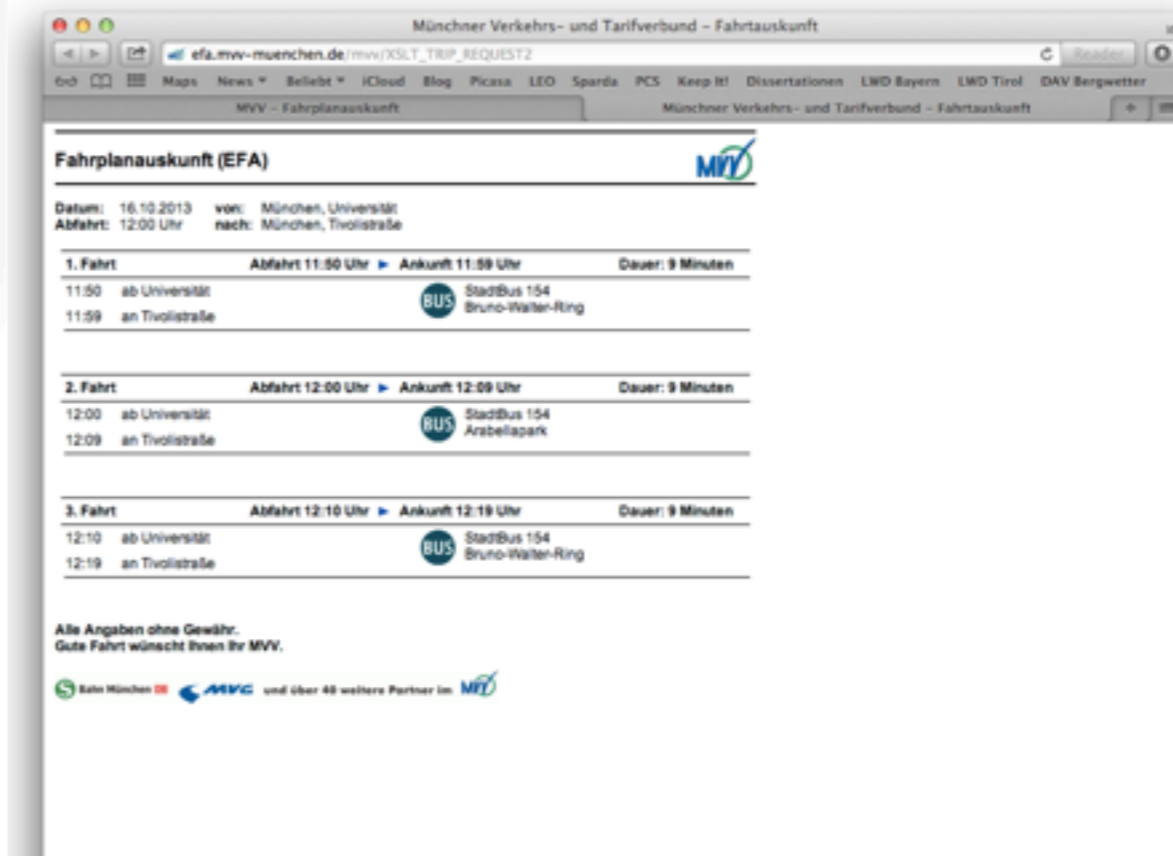
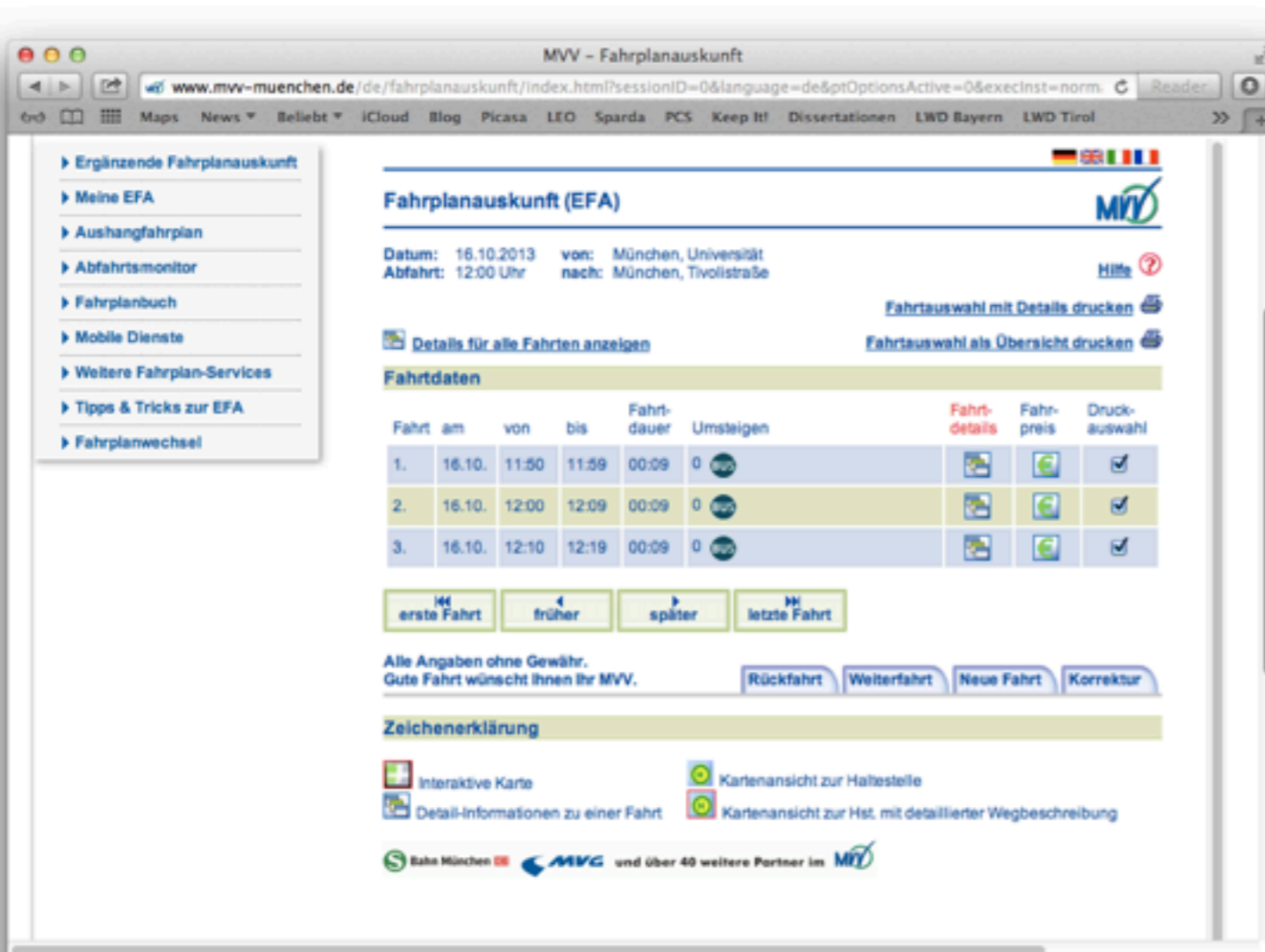
Fluid Web Design



Responsive Web Design



Responsiveness across media



- e.g., screen vs. print
- examples:
 - <http://www.mvv-muenchen.de>
 - <http://www.bahn.de>

Basic Elements of Responsive Design

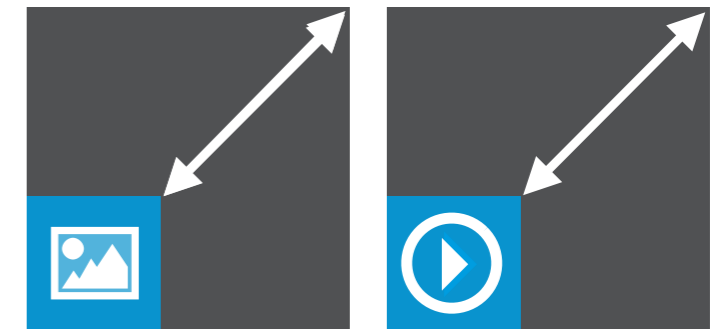
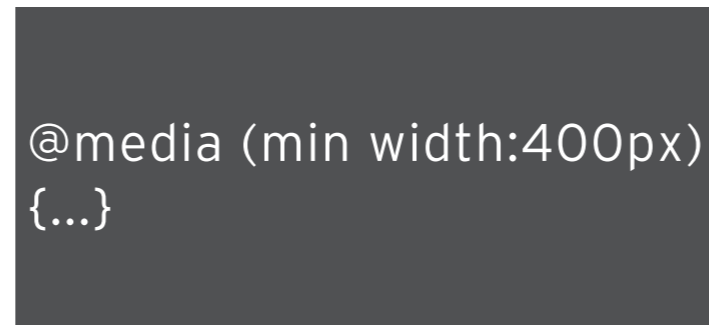
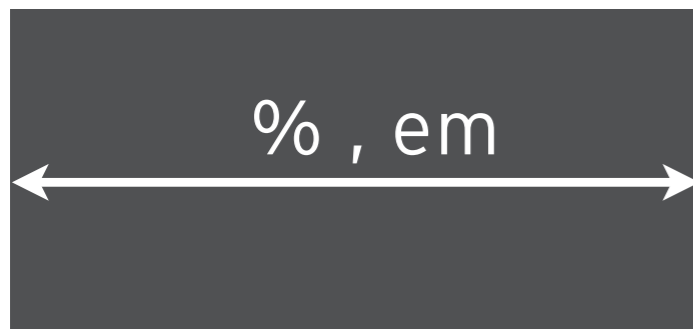
Flexible Grid Layout



Media Queries

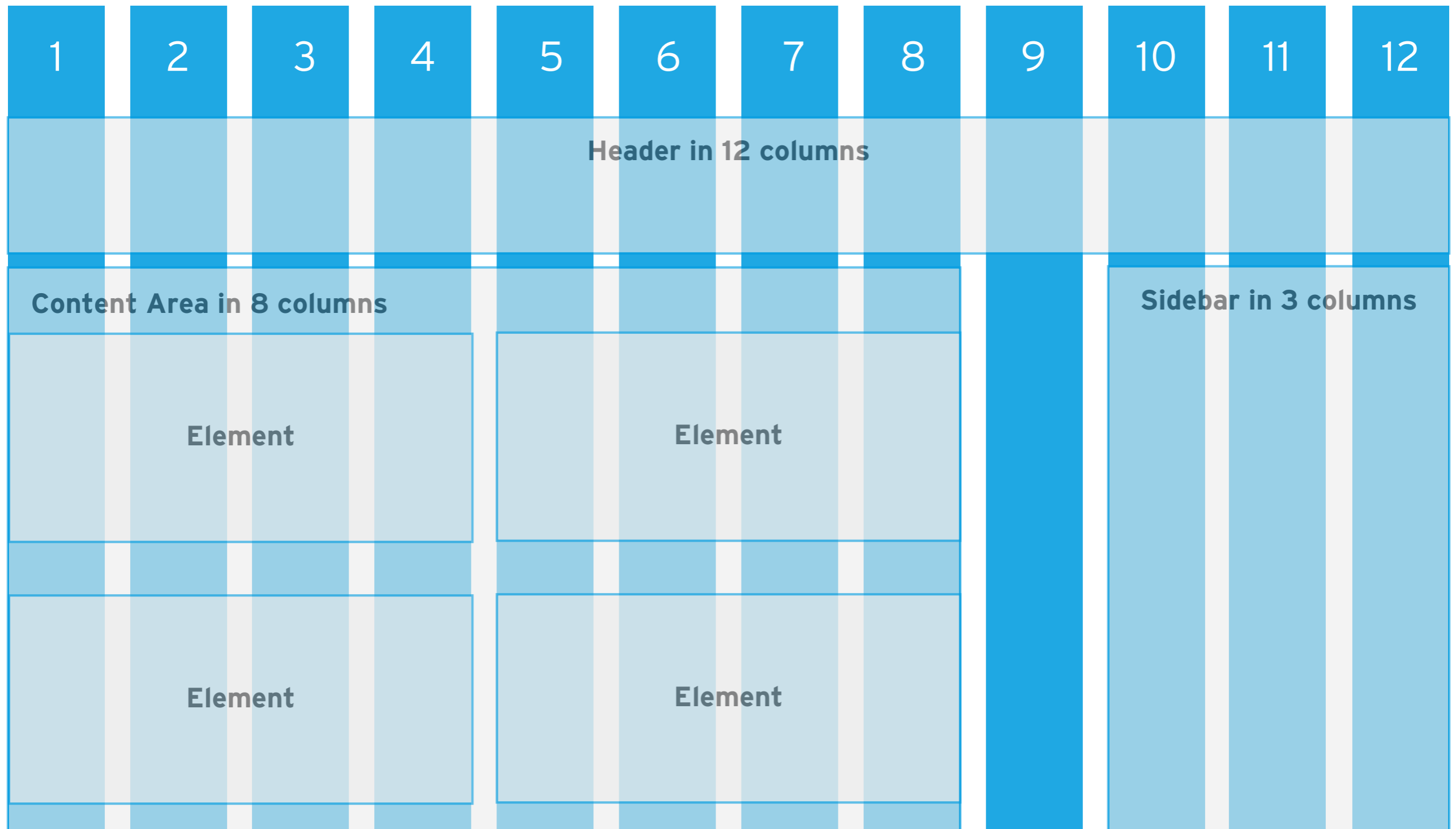


Flexible Images
+ Media



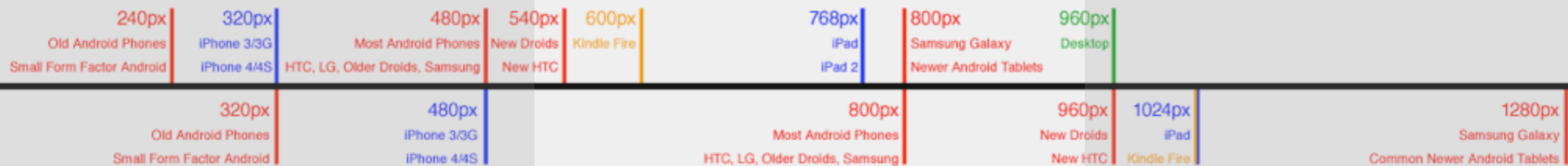
- Source: presentation by Nicolas Leyking
 - <http://de.slideshare.net/ERGOSIGN/vortrag-responsives-design-upa-2013>

Flexible grid layout (example)



Switching between layouts

Device Widths For Responsive Design Planning - March 2012



<http://www.metaltoad.com/blog/simple-device-diagram-responsive-design-planning>

Media Queries

- width/height
- device-width/-height
- orientation
- aspect ratio
- color
- resolution
- medium
 - screen
 - print
 - handheld

```
1  @media screen and (max-device-width: 480px) {  
2      /**  
3          some CSS Code  
4      **/  
5  }
```

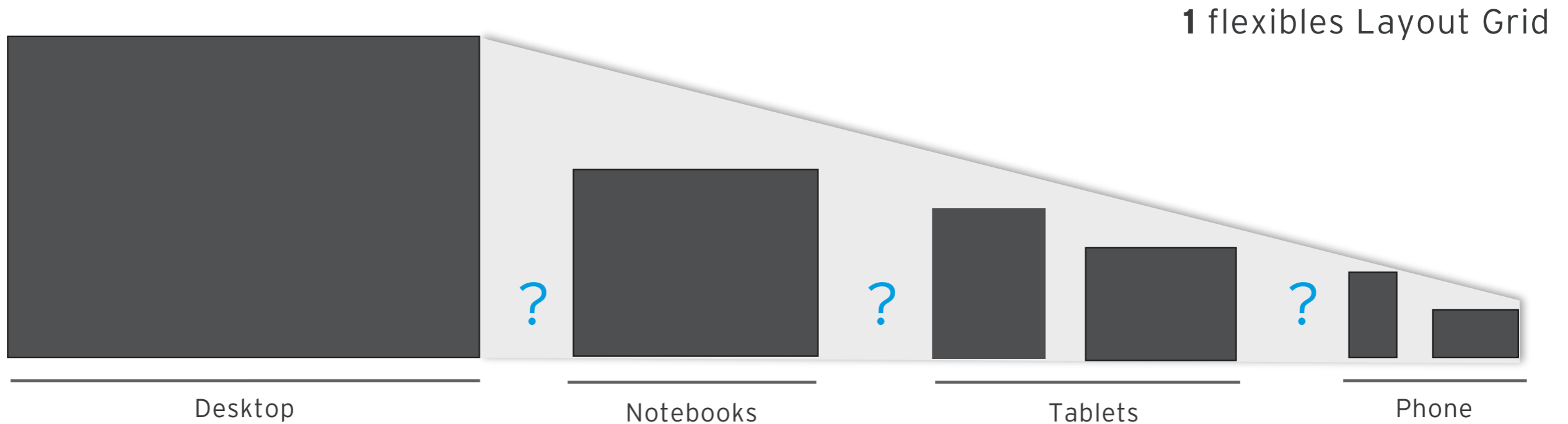
```
1  @media screen and (orientation: portrait) {  
2      /**  
3          some CSS Code  
4      **/  
5  }
```

```
1  @media screen and (device-aspect-ratio: 1280/720) {  
2      /**  
3          some CSS Code  
4      **/  
5  }
```

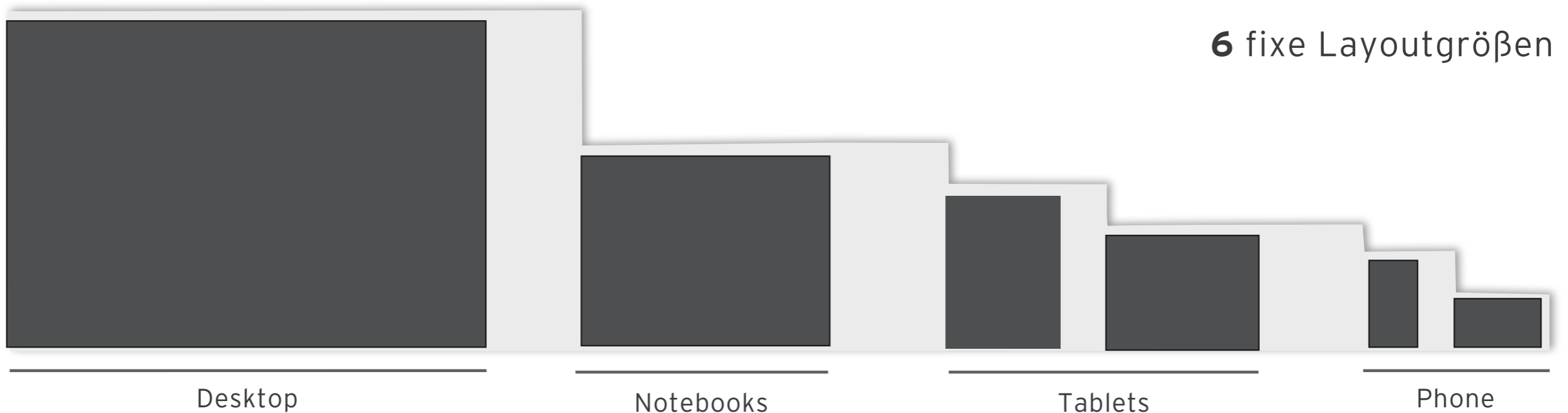
<http://www.w3.org/TR/css3-mediaqueries/>

Responsive vs. Adaptive Design

Responsives Design



Adaptives Design



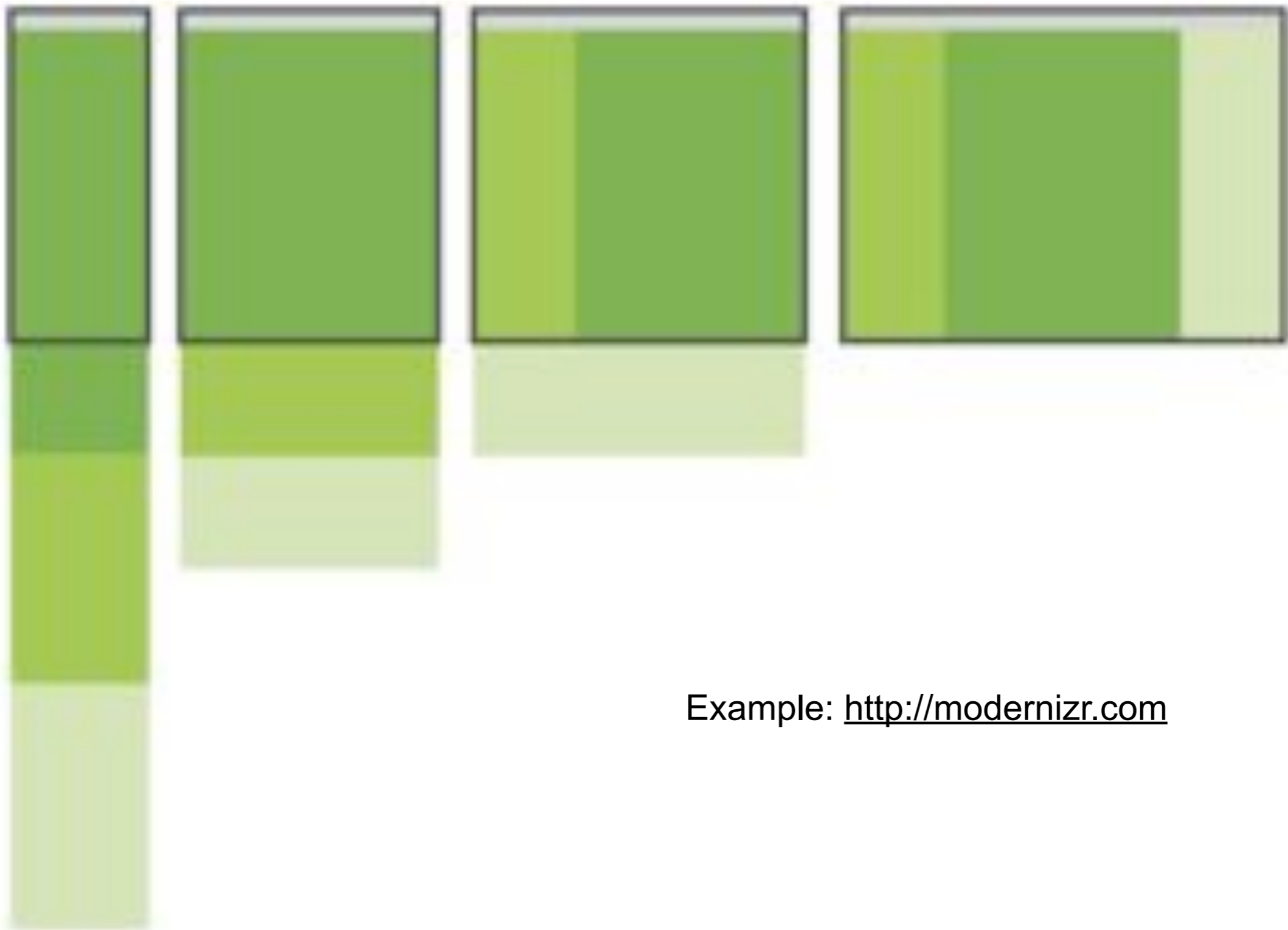
Flexible Image Scaling



Some design patterns for RWD

- Not a comprehensive list, just examples
- Layout patterns
 - column drop
 - off canvas
- Navigation patterns
 - navigation drawer
 - Toggle
- Accommodating touch across different screen sizes
- Sources:
 - presentation by Nicolas Leyking (<http://de.slideshare.net/ERGOSIGN/vortrag-responsives-design-upa-2013>)
 - <http://www.lukew.com/ff/entry.asp?1649>

Layout: Column drop



Example: <http://modernizr.com>

Modernizr: the feature detection library for HTML5/CSS3

modernizr.com

Maps News Beliebt iCloud Blog Picasa LEO Sparda PCS Keep It! Dissertationen LWD Bayern LWD Tirol DAV Bergwetter

DOWNLOAD DOCUMENTATION RESOURCES NEWS

Modernizr
FRONT-END DEVELOPMENT DONE RIGHT

“An indispensable tool.”
— Bruce Bowman, sr. product manager, Edge Tools & Services

Modernizr is a JavaScript library that detects HTML5 and CSS3 features in the user's browser.

Why use Modernizr?

Taking advantage of cool new web technologies is great fun, until you have to support browsers that lag behind. Modernizr makes it easy for you to write conditional JavaScript and CSS to handle each situation, whether a browser supports a feature or not. It's perfect for doing progressive enhancement easily.

How it works

Modernizr runs quickly on page load to detect features; it then creates a JavaScript object with the results, and adds classes to the `html` element for you to key your CSS on. Modernizr supports dozens of tests, and optionally includes [YepNope.js](#) for conditional loading of external `.js` and `.css` resources.

Check out the [full list of features](#) that Modernizr detects, or learn more about [conditional resource loading with Modernizr](#).

Download Modernizr 2.6.2

Use the commented, uncompressed Development version to develop with and learn from.

Then, dive into the Production build tool and pick just the tests you need!

[View documentation](#)

DEVELOPMENT
Uncompressed, 42 Kb

PRODUCTION
Configure Your Build

Get started with Modernizr

While Modernizr gives you finer control over the experience through JavaScript-driven feature detection, it is important to continue to use best practices throughout your development process. Use progressive enhancement wherever you can, and don't sacrifice accessibility for convenience or performance.

- [Documentation: Getting started](#)
- [Taking Advantage of HTML5 and CSS3 with Modernizr](#), Faruk Ateş
- [How to use Modernizr](#), Inayalli de León
- [Modernizr: front-end development done right](#), Ryan Seddon
- [wiki] [The Undetectables: features that cannot be detected](#)
- [wiki] [Cross-browser Polyfills](#)

Also check out our [Resources section](#).

Tip: check our [Modernizr test suite](#) to quickly test your current browser's features.

Follow us on Twitter

Contribute on GitHub

Subscribe with RSS

Donate to Modernizr

Latest news

April 2nd, 2013
[Modernizr 3, Stickers & Diversity](#)
News on v.3, stickers, and a message from the team.

PREVIOUS

January 7th, 2013
[Modernizr Policy on Browser Bugs and Feature Detects](#)

September 4th, 2012
[Modernizr 2.6.2 released](#)

July 19th, 2012
[Modernizr 2.6 released](#)



DOWNLOAD DOCUMENTATION RESOURCES NEWS

“An indispensable tool.”

— Bruce Bowman, sr. product manager, Edge Tools & Services

Modernizr is a JavaScript library that detects HTML5 and CSS3 features in the user's browser.

Why use Modernizr?

Taking advantage of cool new web technologies is great fun, until you have to support browsers that lag behind. Modernizr makes it easy for you to write conditional JavaScript *and* CSS to handle each situation, whether a browser supports a feature or not. It's perfect for doing progressive enhancement easily.

How it works

Modernizr runs quickly on page load to detect features; it then creates a JavaScript object with the results, and adds classes to the `html` element for you to key your CSS on. Modernizr supports dozens of tests, and optionally includes [YepNope.js](#) for conditional loading of external `.js` and `.css` resources.

Check out the [full list of features](#) that Modernizr detects, or learn more about [conditional resource loading with Modernizr](#).

Download Modernizr 2.6.2

[View documentation](#)

Use the commented, uncompressed Development version to develop with and learn from.

 **DEVELOPMENT**
Uncompressed, 42 Kb

Then, dive into the Production build tool and pick just the tests you need!

 **PRODUCTION**
Configure Your Build

Get started with Modernizr

While Modernizr gives you finer control over the experience through JavaScript-driven feature detection, it is important to continue to use best practices throughout your development process. Use progressive enhancement wherever you can, and don't sacrifice accessibility for convenience or performance.

- [Documentation: Getting started](#)
- [Taking Advantage of HTML5 and CSS3 with Modernizr](#), Faruk Ateş
- [How to use Modernizr](#), Inayalli de León
- [Modernizr: front-end development done right](#), Ryan Seddon
- [wiki] [The Undetectables: features that cannot be detected](#)
- [wiki] [Cross-browser Polyfills](#)

Also check out our [Resources section](#).

Tip: check our [Modernizr test suite](#) to quickly test your current

The image shows a browser window displaying the Modernizr website. The browser's address bar shows 'modernizr.com' and the page title is 'Modernizr: the feature detection library for HTML5/CSS3'. The website has a dark navigation bar with links for 'Download', 'Documentation', 'Resources', and 'News'. The main content area has a pink background with a dark banner at the top containing the Modernizr logo. Below the banner, there is a quote from Bruce Bowman, sr. product manager at Edge Tools & Services, describing Modernizr as an indispensable tool. The main text explains that Modernizr is a JavaScript library for detecting HTML5 and CSS3 features. It includes sections for 'Why use Modernizr?' and 'How it works', which describe how the library detects features and adds classes to the HTML element. At the bottom, there is a dark box with the text 'Download Modernizr 2.6.2' and instructions on how to use the development and production versions.

Modernizr: the feature detection library for HTML5/CSS3

modernizr.com

Download Documentation Resources News

Modernizr

“An indispensable tool.”
— Bruce Bowman, sr. product manager, Edge Tools & Services

Modernizr is a JavaScript library that detects HTML5 and CSS3 features in the user’s browser.

Why use Modernizr?

Taking advantage of cool new web technologies is great fun, until you have to support browsers that lag behind. Modernizr makes it easy for you to write conditional JavaScript *and* CSS to handle each situation, whether a browser supports a feature or not. It’s perfect for doing progressive enhancement easily.

How it works

Modernizr runs quickly on page load to detect features; it then creates a JavaScript object with the results, and adds classes to the `html` element for you to key your CSS on. Modernizr supports dozens of tests, and optionally includes [YepNope.js](#) for conditional loading of external `.js` and `.css` resources.

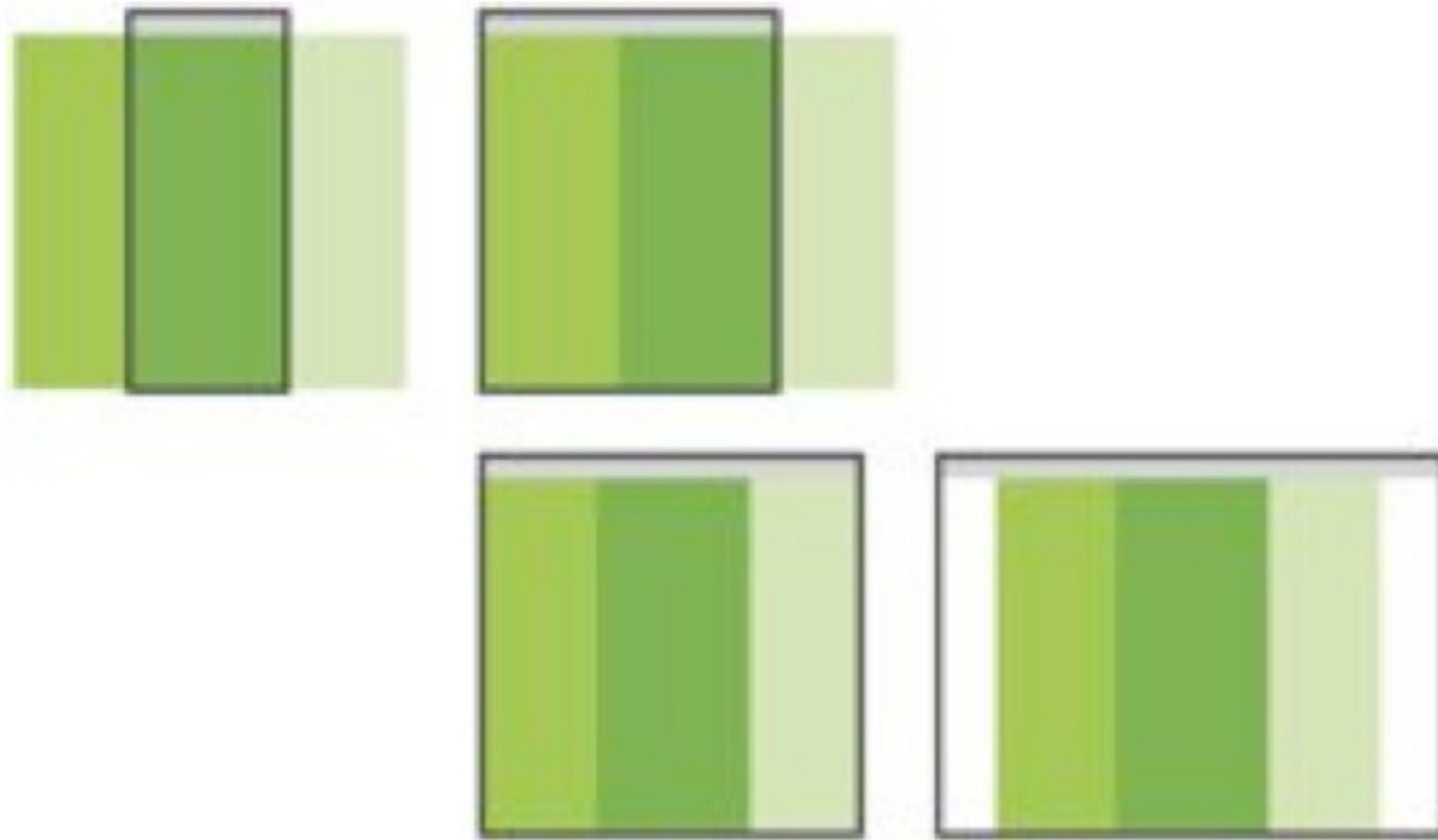
Check out the [full list of features](#) that Modernizr detects, or learn more about [conditional resource loading with Modernizr](#).

Download Modernizr 2.6.2

Use the commented, uncompressed Development version to develop with and learn from.

Then, dive into the Production build tool and pick just the tests you need!

Layout: Off Canvas



Example: <http://www.kaemingk.com/de/>



Home | Kaemingk.com

www.kaemingk.com/de/

Maps News ▾ Beliebt ▾ iCloud Blog Picasa LEO Sparda PCS Keep It! Dissertationen LWD Bayern LWD Tirol

Modernizr: the feature detection library for HTML5/CSS3 Home | Kaemingk.com

KONTAKT DEUTSCH ▾

KAEMINGK
season decorations

HOME KOLLEKTIONEN SHOWROOM MARKEN AGENDA/NEUIGKEITEN ÜBER KAEMINGK



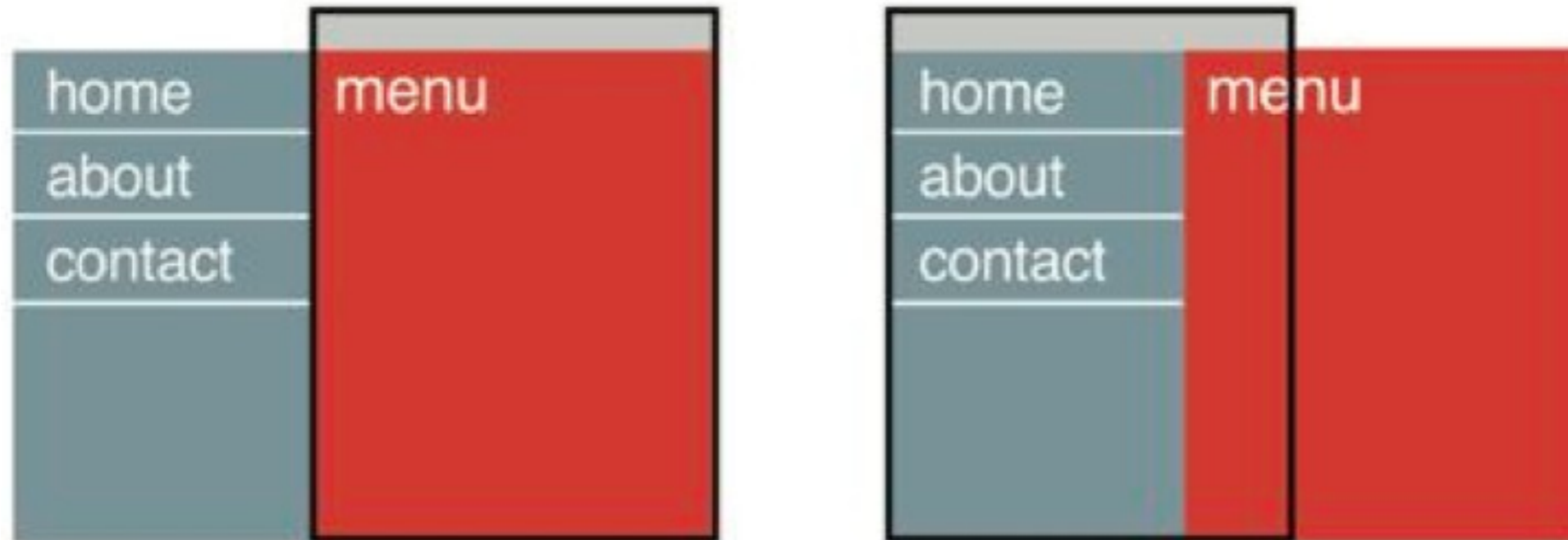
Finden Sie Ihre
Frühjahrs-/Sommer-Inspiration
bei Kaemingk

ERHALTEN SIE EINEN EINDRUCK UNSERER
Frühling 2014
KOLLEKTION

BESICHTIGEN
SIE UNSEREN
SHOWROOM



Navigation Drawer



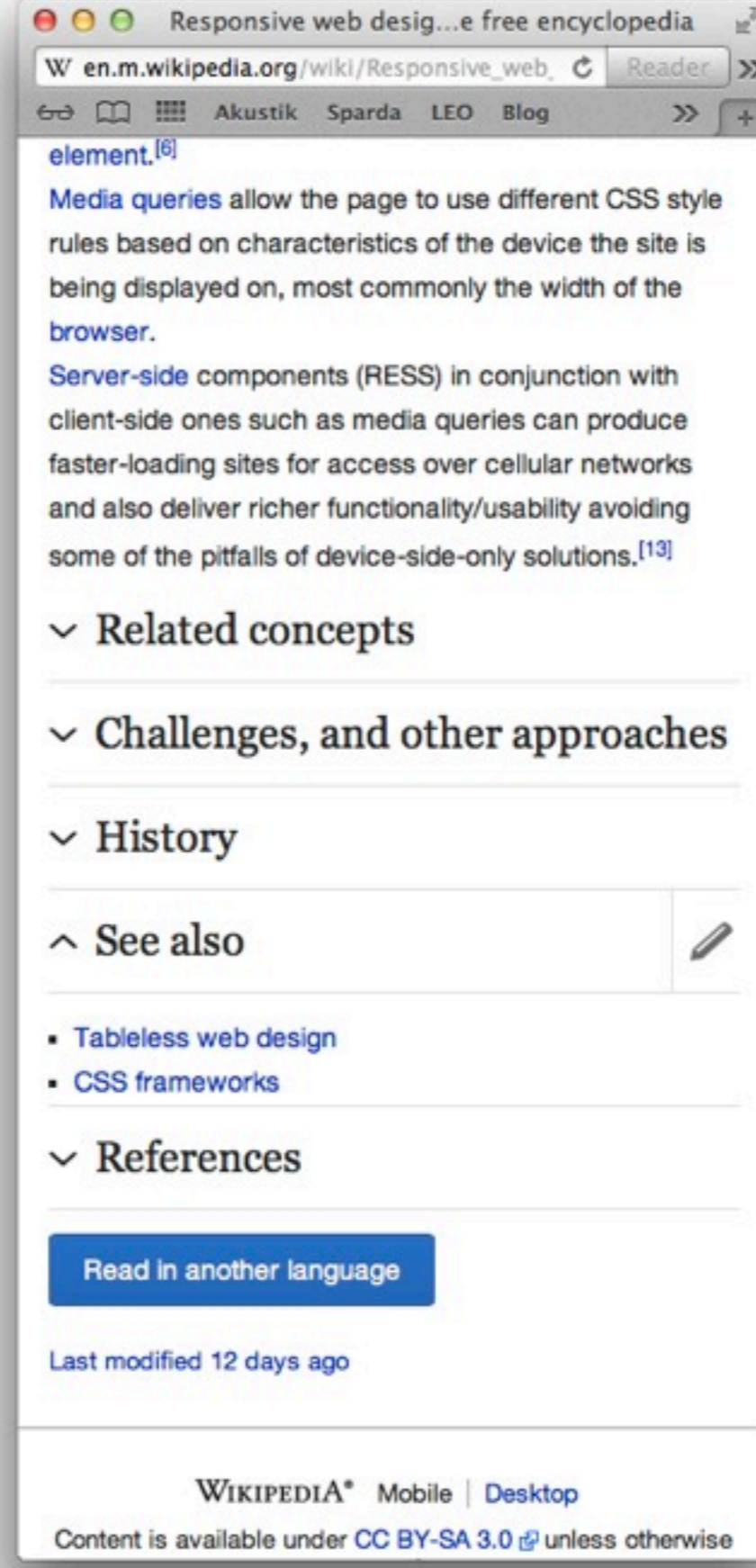
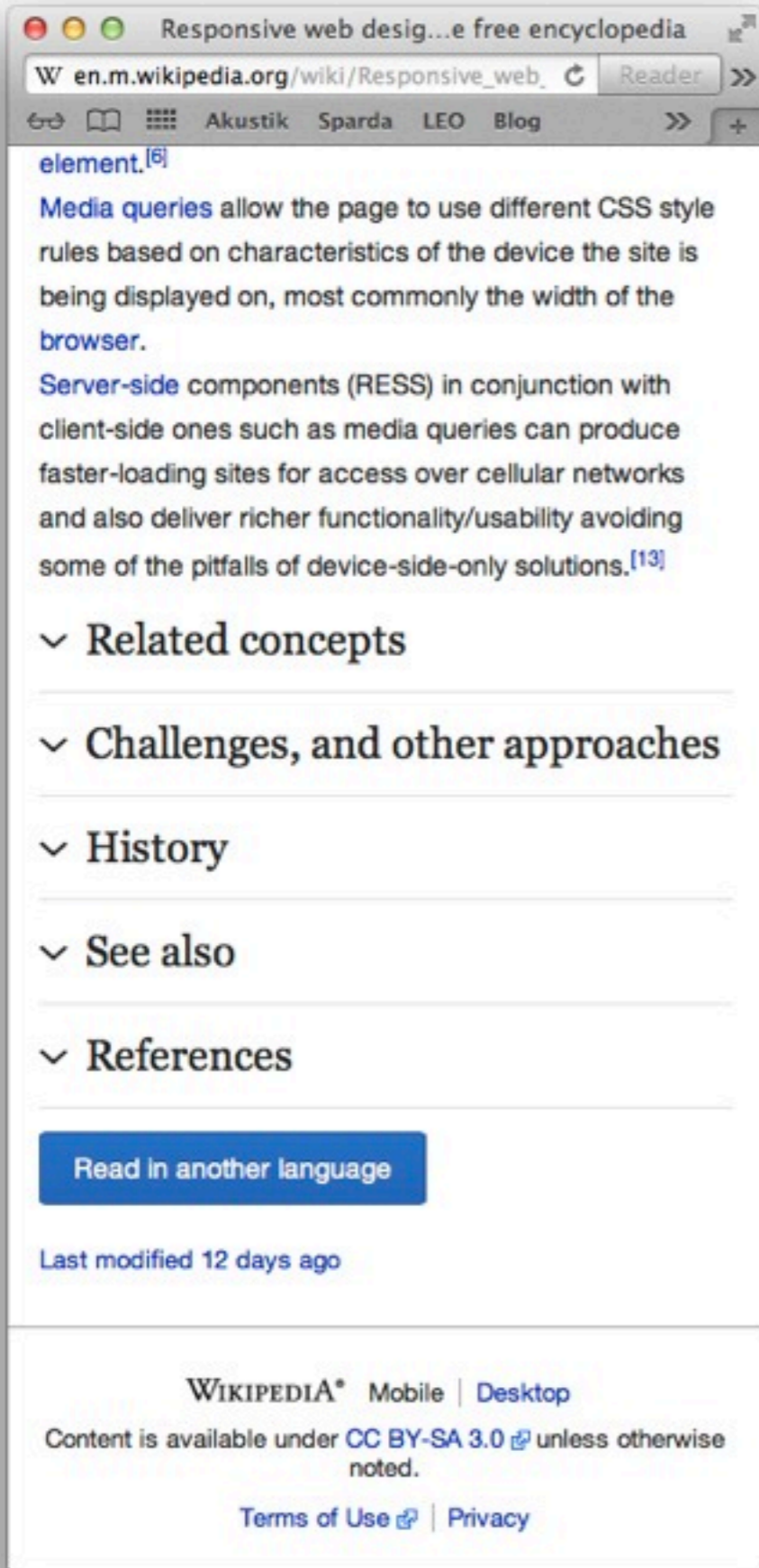
Example: <http://www.kaemingk.com/de/>



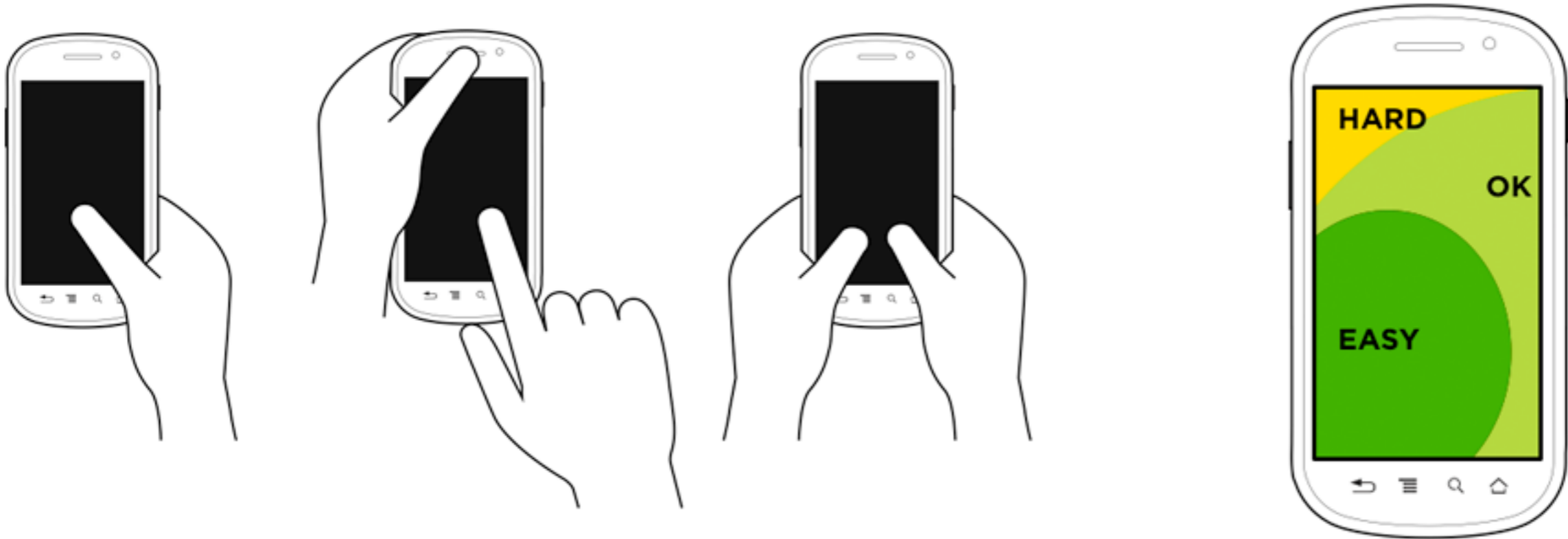
Toggle



Example: Wikipedia mobile



Touch on mobile phones (<http://www.lukew.com/ff/entry.asp?1649>)



- bottom of screen is easiest to reach
- top corner is hardest to reach
- we don't know how the device will be held
- put important stuff at the bottom

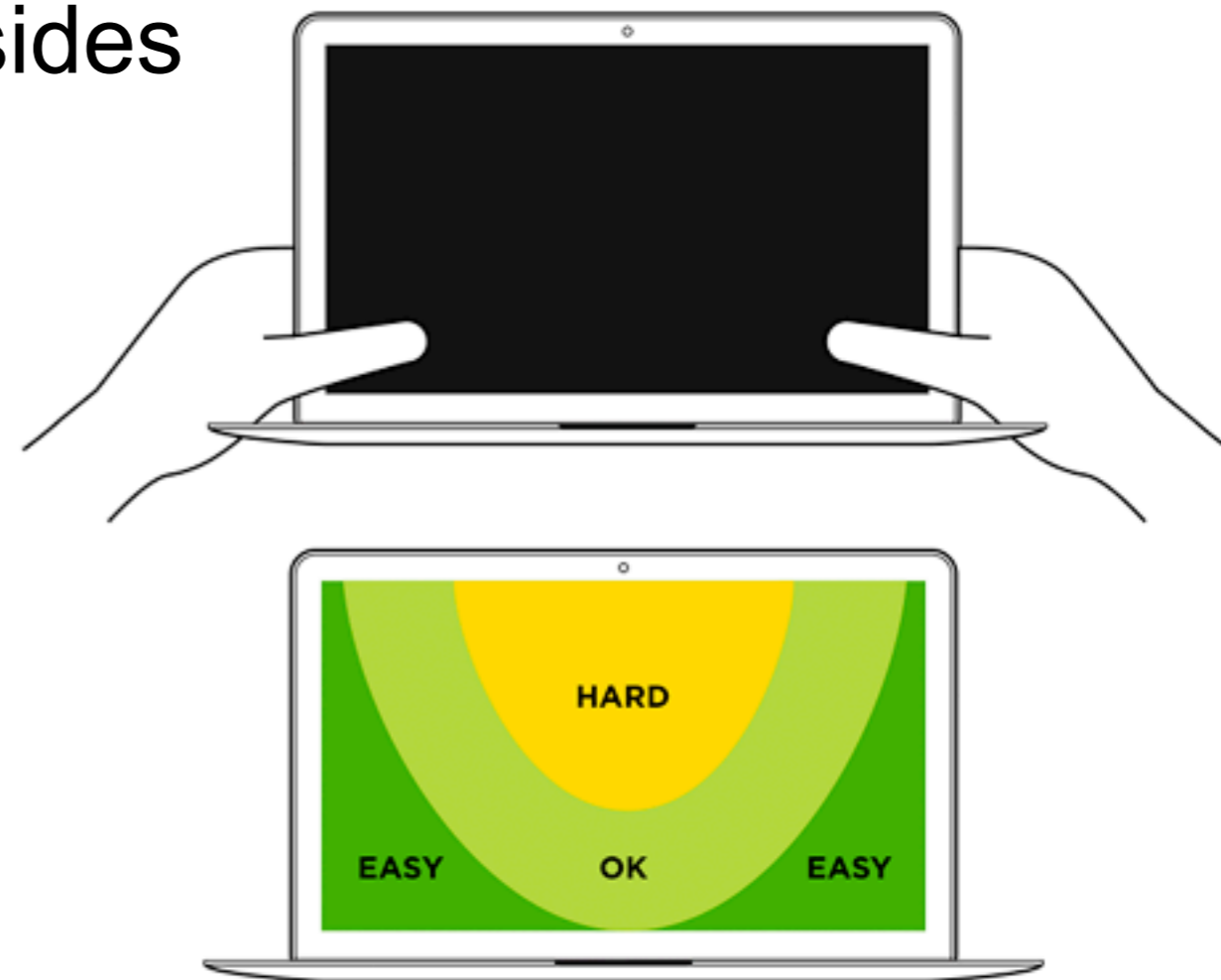
Touch on tablets (<http://www.lukew.com/ff/entry.asp?1649>)



- bottom corners are easiest to reach
- top middle is hardest to reach
- again: we don't know how exactly the device will be held
- put important stuff at the bottom and to the sides

Touch on ultrabooks (<http://www.lukew.com/ff/entry.asp?1649>)

- bottom corners are easiest to reach
- top middle is hardest to reach
- not so many ways to hold the device
- put important stuff along the sides



Adaptive Layout patterns (<http://www.lukew.com/ff/entry.asp?1649>)

- Mobile: usually OK
- Larger layouts:
 - designed for mouse
 - wrong areas for touch
 - discriminate between desktop and handheld!
 - adapt touch areas to tablet usage if possible!



HCI and the Web

- Organizational stuff
- Usability: a quick reminder
- How we use the web (device side)
- How we use the web (human side)
- HTML+CSS: a quick reminder
- Responsive Web Design: some Basics
- Responsive Web Design: Example and Exercise

Resources for self-teaching and reference

- <http://www.w3.org>
- <http://www.selfhtml.org>

- <http://www.csszengarden.com>
- <http://css-tricks.com>

- <http://coding.smashingmagazine.com/2011/01/12/guidelines-for-responsive-web-design/>
- <http://www.lukew.com/ff/entry.asp?1649>

- <http://getbootstrap.com/>
- <http://www.mobilephoneemulator.com>

Task for the exercise

- Build a responsive website for the Olyphonics Choir
 - web site should work on desktop, tablet, phone + print
 - should be easily maintainable (i.e. without complicated tools)
- Structure is given as a result of initial discussions
- Text is given in a .doc file (right from the texter ;-)
- Photo currently is just a placeholder
 - better photo will be shot on Oct. 26th
 - provide room for a gallery of future photos
 - provide room for future announcements
- Best solution will actually be used
 - author will be acknowledged on the web page

Just the end of the process

- Requirements
 - Web site structure from group discussions
 - existing logo should be used
 - domain <http://olyphonics.de> is already blocked
- Initial design
 - Infos collected and text formulated
 - technology decision: HTML5 + CSS3
 - decision for a responsive design (potentially single page?)
- Initial functional prototype
- Design iterations
 - defining visual and functional design
 - tests on different platforms

Base material for the exercise

- <http://www.cip.ifi.lmu.de/~butz/olyphonics/website/>
- <http://www.cip.ifi.lmu.de/~butz/olyphonics/website/Homepage.docx>
- <http://www.cip.ifi.lmu.de/~butz/olyphonics/website/olylogo-big.jpg>
- http://www.cip.ifi.lmu.de/~butz/olyphonics/website/D2X_2357_small.jpg

Appendix: Potential Problems on the Web

- Downright errors
 - Annoying or inaccessible page design
 - Search engine problems
 - Information architecture problems
 - E-commerce problems
-
- From <http://www.siteusability.com/mistakes.html> (now offline)-:

What are potential problems? (1)

- Downright errors:
 - Broken links or missing images.
 - Firewall errors, server cannot be contacted, directory browsing not allowed (or allowed?).
 - Scripting errors that pop up an error message, make the page unusable, or write strings of gibberish amongst the text.
 - HTML coding errors that mean the page doesn't display properly, or at all.

What are potential problems? (2)

- Annoying or inaccessible page design:
 - An "entrance tunnel" or splash screen - lots of flashy imagery but no real content that requires a click to get to the real home page.
 - Pages with such poor contrast between background and text they are hard to read.
 - Text in tiny or illegible fonts.
 - Pages that take minutes to download (even worse if when they have finished, you weren't interested in the content anyway).
 - Content that requires a specialised plug-in to read it.
 - Pages that require a specific browser to display nicely.
 - Links that lead to "under construction" pages.
 - Link colour schemes where you can't tell which ones you have already visited.
 - Links with badly-chosen targets that display numerous hidden windows on the desktop, break the Back button, or display pages without the necessary menus to use them properly.
 - Forms where you don't know what the site owners want to do with the information you are asked to supply.
 - Forms that don't explain properly what you need to enter, or don't let you go back and amend any errors.
 - Pages with typographical or grammatical errors, confusing and poorly-written text, or inconsistent terminology.

What are potential problems? (3)

- Search engine problems:
 - Pages with no links to other pages in the site.
 - Pages called "No title", "Untitled", "Insert document title here", and/or with a meaningless abstract, so the user has no idea whether the link is relevant or not.
 - Pages that no longer exist on your site because you moved or renamed them.
 - Pages so poorly designed they will never even appear in a search engine listing.

What are potential problems? (4)

- Information architecture problems:
 - Pages with different layouts and appearance for the same kind of information.
 - Very long pages with no quick way to skip about them.
 - Forms that don't work in a comprehensible way, and shopping cart systems that confuse in their complexity.
 - Links that lead to mystery destinations (e.g. "click here"), or to other sites without warning.
 - Overwhelming numbers of links on the home (or other) page.
 - Menu options or navigation bar icons that mean little to the average visitor.
 - No consistent way to move around the site on every page.
 - No clear distinction between different kinds of information.
 - Confusing site structure so the visitor cannot guess where to go for information.

What are potential problems? (5)

- E-commerce problems:
 - Potential buyers can't find the product they want because they don't understand the categories you have chosen.
 - Visitors leave without purchasing because they don't want to register.
 - Visitors can't find your returns policy or how their privacy is protected if they buy from you.
 - Buyers have to work out the shipping and handling charges for themselves when viewing an item in your online catalogue.
 - Visitors from overseas don't understand the measurement system you use for sizes or weights.
 - ... the list of potential problems is endless - this just skims the surface for sites selling to the consumer.

Appendix: Dos and don'ts

from <http://de.slideshare.net/ERGOSIGN/vortrag-responsives-design-upa-2013>

+ Dos

- + Sorge für ein einheitliches, teamweites Verständnis über Responsivität
- + Sammel soviel Informationen wie möglich über deine Zielplattformen
- + Teste dein responsives Design so früh wie möglich mit leichtgewichtigen Prototypen
- + Sammel Feedback von langjährigen Benutzern der Plattformen
- + Dokumentiere das Layoutverhalten durch anschauliche Mittel
- + Stelle die Dokumentation und die Prototypen den Entwicklern zur Verfügung

- Don'ts

- Einsatz von fixen layouting Tools (Photoshop)
- Entwicklungsbeginn vor Abschluss und Abnahme des Designs
- 1:1 Übertragung des Designs von Desktop auf Mobile
- Verzicht auf relevante Inhalte aufgrund von Platzmangel.