Универзитет „Св. Кирил и Методиј“ - Скопје
ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ
И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО

Lesson 1

Introduction

Web Design

February 14, 2014

Course Informations

Professors:

Ivan Chorbev, PhD - ivan.chorbev@finki.ukim.mk
Goce Armenski, PhD - goce.armenski@finki.ukim.mk

Assistants

Boban Joksimoski - boban.joksimoski@finki.ukim.mk
Ivan Kitanovski -ivan.kitanovski@finki.ukim.mk

course materials and news:

http://courses.finki.ukim.mk/

What will we learn in this course?

- Design Theory - for usage in WWW
- Usage of basic techniques and technologies for web design
- Basic concepts of graphic design
- **HTML** - its the method for representing data on web pages. Currently in general usage is HTML 4.x, but HTML5 is gaining a lot of support.
- **CSS** - The de-facto method for "designing" web pages. As with HTML, the widespread version is CSS 2.1, but CSS 3 is used a lot.
- Depending on the occasion:
  - JQuery - The JavaScript for traversing the Document Object Model (DOM)
  - Modernizr - detecting of browser HTML5 fetures.

What are you supposed to learn here

- **"No Programming"**, and we will be trying hard to avoid it.
- Learing of basic principles of design, and their usage for creating web pages.
- Acquainting with the widespread standards for creating web pages.
- How should the data be presented to the users!

Web Applications

What should you know?

- **Graphic Design**
- **Interface Design**
- **HTML, CSS**
- Information Design
- Graphic production
- Programming languages
- Multimedia applications

Basic concepts of the Internet

- Internet is a network of interconected networks, that share data and information
- World Wide Web (WWW) is part of the Internet that is concerned with displaying HTML documents and their serving to the users
- WWW standards are responsibility of the **W3C**
- **NOTE:** The "standards" can be ignored.

Accessing Web Pages

- Access to the web pages is by specifying the URL (Uniform Resource Locator).
- URL is a subset of URI (Uniform Resource Identifier), that is specified in RFC 3986
- General syntax of an URI:
  `scheme ":" authority ["/" path ] ["?" query ] ["#" fragment ]`
- The URI is consisted of mandatory and optional parts (text in brackets - `[]`).
  examples:
  `ftp: // joksim@server.finki.ukim.mk / data/`
  `https :// mail.google.com / mail/u/0/ ? ui=2&shva=1 # spam`

Meaning of the different URI parts

Parts of an URI

- **Scheme**
  is a specification for assigning identifiers in an URI. There are a lot of schemes, both registered and individual, but we'll be using the common schemes: http, https, ftp, etc. It is specified from the user or from the client tool.
- **Authority**
  enables the contact with the requested location. The start of the authority is defined after the "//". The parts of the authority are:

```
[userinfo"@"]host[":"port]
```

Meaning of the different URI parts

Parts of an URI

- **Path**
  is the path where the requested data is saved on the filesystem. It begins with the first slash "/" and ends with the end of the URI or the begining of the query or fragment.
- **Query**
  is consisted of one or more key=value parts, separated with "&". The query part begins with the character "?" and ends with the URI or begining of the fragment.
- **Fragment**
  is a part responsible for secondary navigation. It starts from the character "#" and ends with the URI.

Why do we mention URI

- The way users access the websites is important!!!
- There is no "rule" for correctly designed URI-s.
- Can access non-local resources (not on the machine).
- Users want to know the resourse that they are accessing and the way they are doing it.
- If the designers don't handle the URI-s, than how can users?

How are web pages "delivered"?
For "static" websites (1)

Static sites do not process the data from clients and the information are "same" for all users (all of the users are viewing the same page)

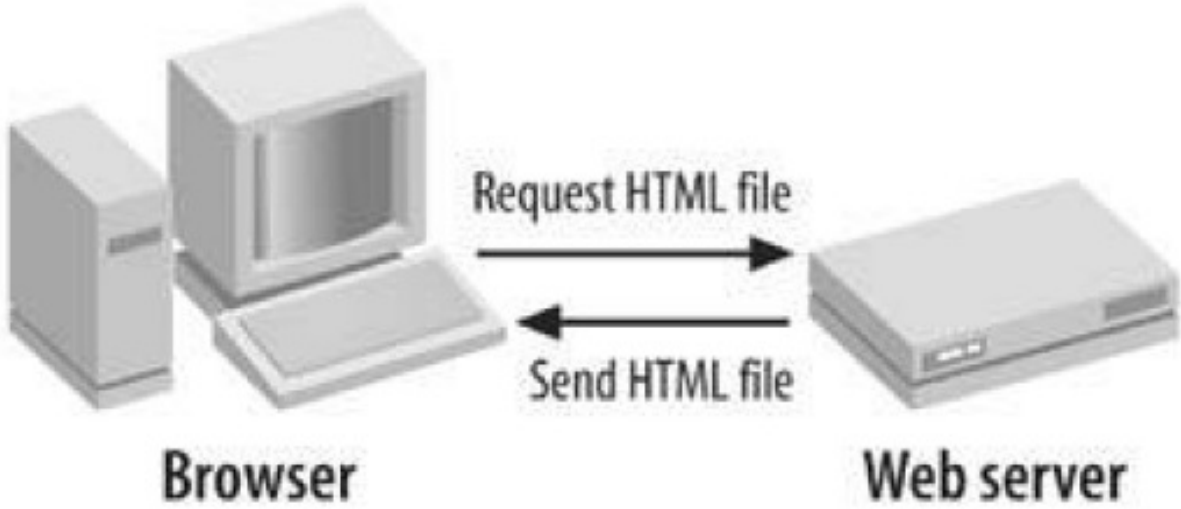
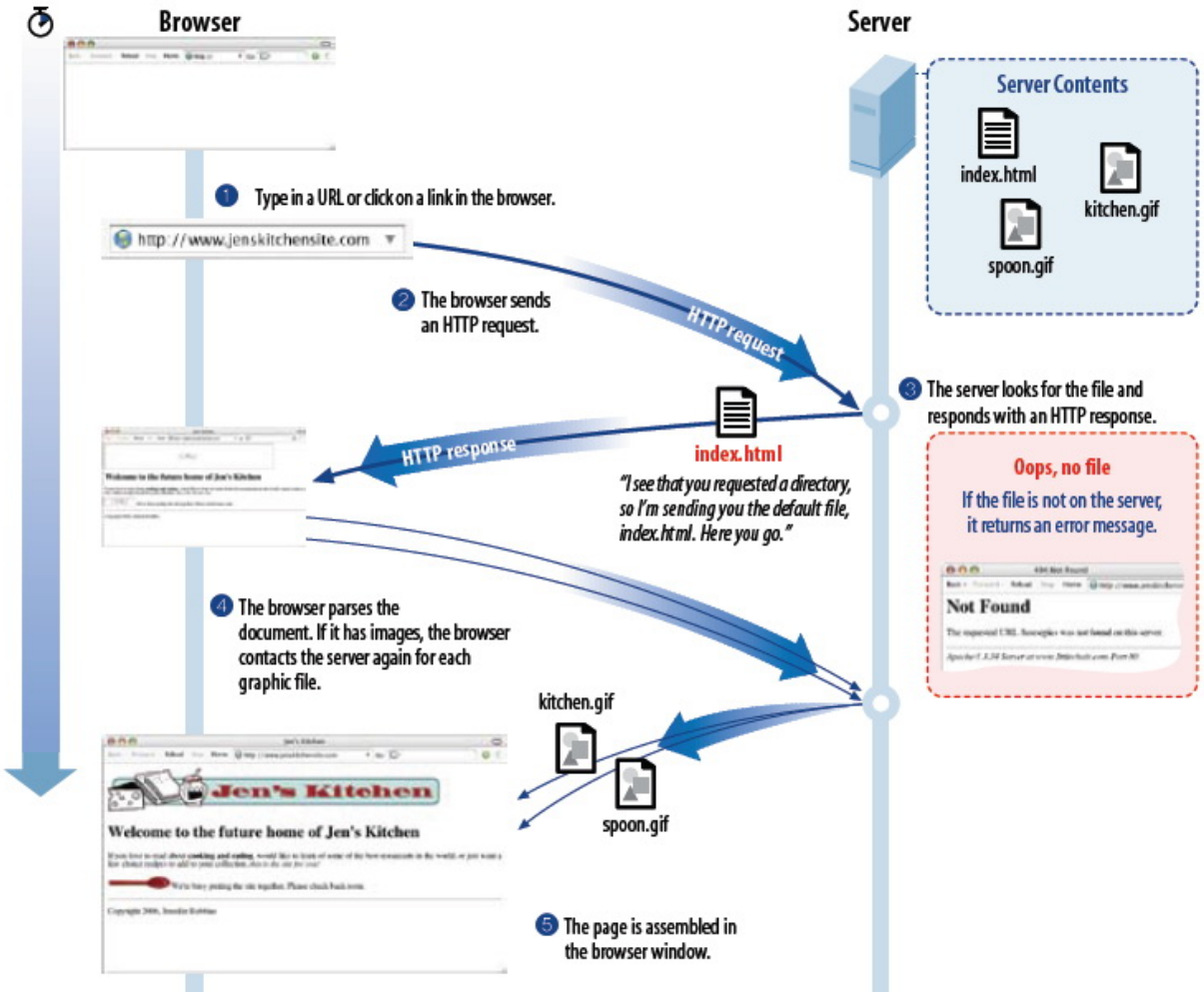


Figure : Serving of static website

How are web pages "delivered"?
For "static" websites (2)



How are web pages "delivered"?
For "dynamic" websites

Web pages are generated individually for every user and the displayed data is different.

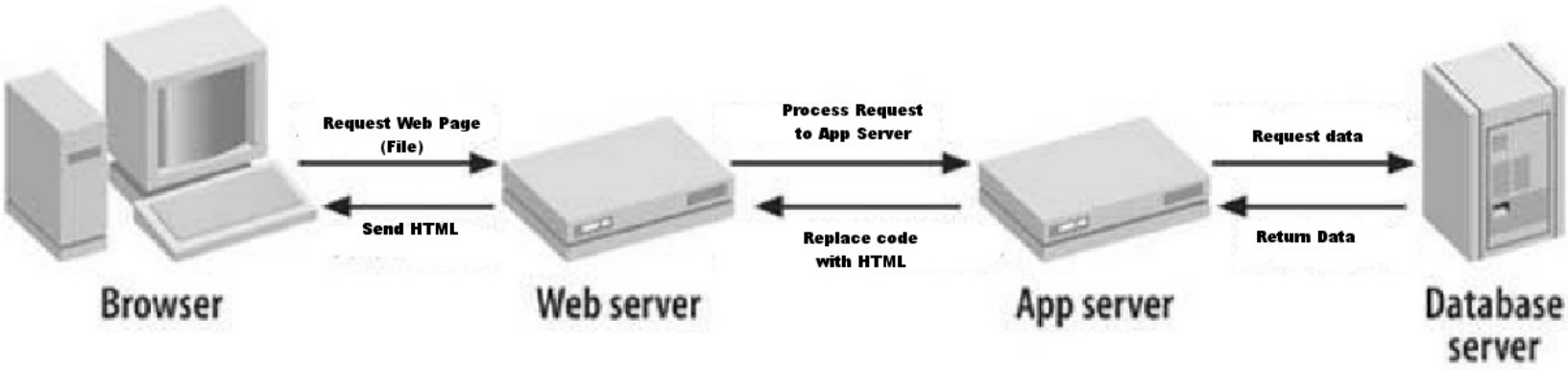


Figure : Serving of dynamic websites

Servers and hosting

Under web server we can consider the hardvare and software needed for serving the users. From hardware aspect, web server is every computer that has an internet connection. From software perspective, its a computer that has installed web server software.

Required software for web server::

- **static website** - web server + internet connection
- **dynamic website** - web server + internet connection
  - application server (can be a separate machine)
  - database server (usually is a separate machine)

Web Servers

The task for web servers is to process the request from the client, and serve the static page, or if the page is dynamic, to invoke the application server.

The web server does not affect the design of the web pages.

product	hosted web pages	market share %
Apache HTTP Server	359,441,468	53.42%
Microsoft IIS	112,303,412	16.69%
nginx	104,411,087	15.52%
Google Web Server	23,029,260	3.42%

Table : Web servers and market share (may 2013)

What does web design do?

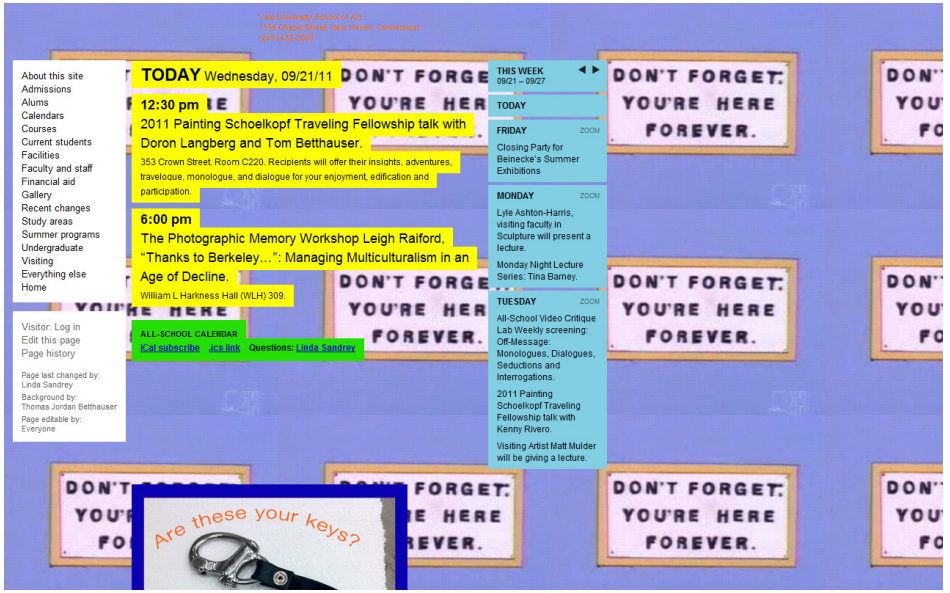
The first and basic goal of web design is to enhance the "experience" of the user.

A web page is consisted of text, pictures, multimedia data and interactive elements.

ALL elements should compose a experience that is aesthetically beautifull.

Example for bad web design!

"Improved" version :)



Tools for web design

Working with HTML and CSS

- WYSIWYG tools (What You See Is What You Get)
  - Adobe Dreamveawer
  - Microsoft Visual Studio
- Text editors and programmer tools
  - Aptana Studio, NetBeans, ...
  - Notepad++, UltraEdit, Bluefish, Emacs, Vim, TextMate, ...
- Software that exports to HTML
  - Microsoft Word, Open Office, ...

Tools for web design

Images

- Raster Images
  - Adobe Photoshop (the de-facto standard)
  - GIMP (Open Source Photoshop alternative)
  - Paint.Net
  - and many others...
- Vector Images
  - Adobe Illustrator (the de-facto standard)
  - CorelDraw (AI alternative)
  - Inkscape (Open Source alternative)
  - and many others...

Tools for web design

Plug-ins

- Can be various digital data like: animations, audio, video, games...
- Separate files.
- Need to use appropriate plug-ins
- Althouth plugins are free, if the user doesn't have them, he will be redirected to get them. He may never return...
- With new technologies like HTML5, it is expected that the plugins will become redundant

Tools for web design

List of common plug-ins

формат	тип	додаток	екст.
Adobe (Macromedia) Flash	Animation, Video	Flash Shockwave Player	swf, flv
Apple QuickTime	Video, Audio	QuickTime Plugin	mov
Microsoft Windows Media	Video, Audio	Windows Media Player	wmv, wma
Real Media	Video, Audio	Real Player	rm, ra
Adobe Acrobat	rich document	Acrobat Reader	pdf
SilverLight (MoonLight)	Video, Audio, Rich Aplication	Silverlight, Moonlight	xap

Table : Commonly used plug-ins

Tools for web design

Web Browsers

Different web browsers:

- Mozilla Firefox
- Microsoft Internet Explorer
- Google Chrome / Apple Safari / Opera
- Konqueror, Netsurf, Elinks, w3m, Lynx - if you have free tine (designer and free time???)

Web Inspectors (Firebug, Developer Tools, Webkit Inspector, Dragonfly).
Other tools (Web Developer, Measure Tools, JavaScript Debugger, . . .).
Internet (Let Me Google That For You) and a lot of patience :)

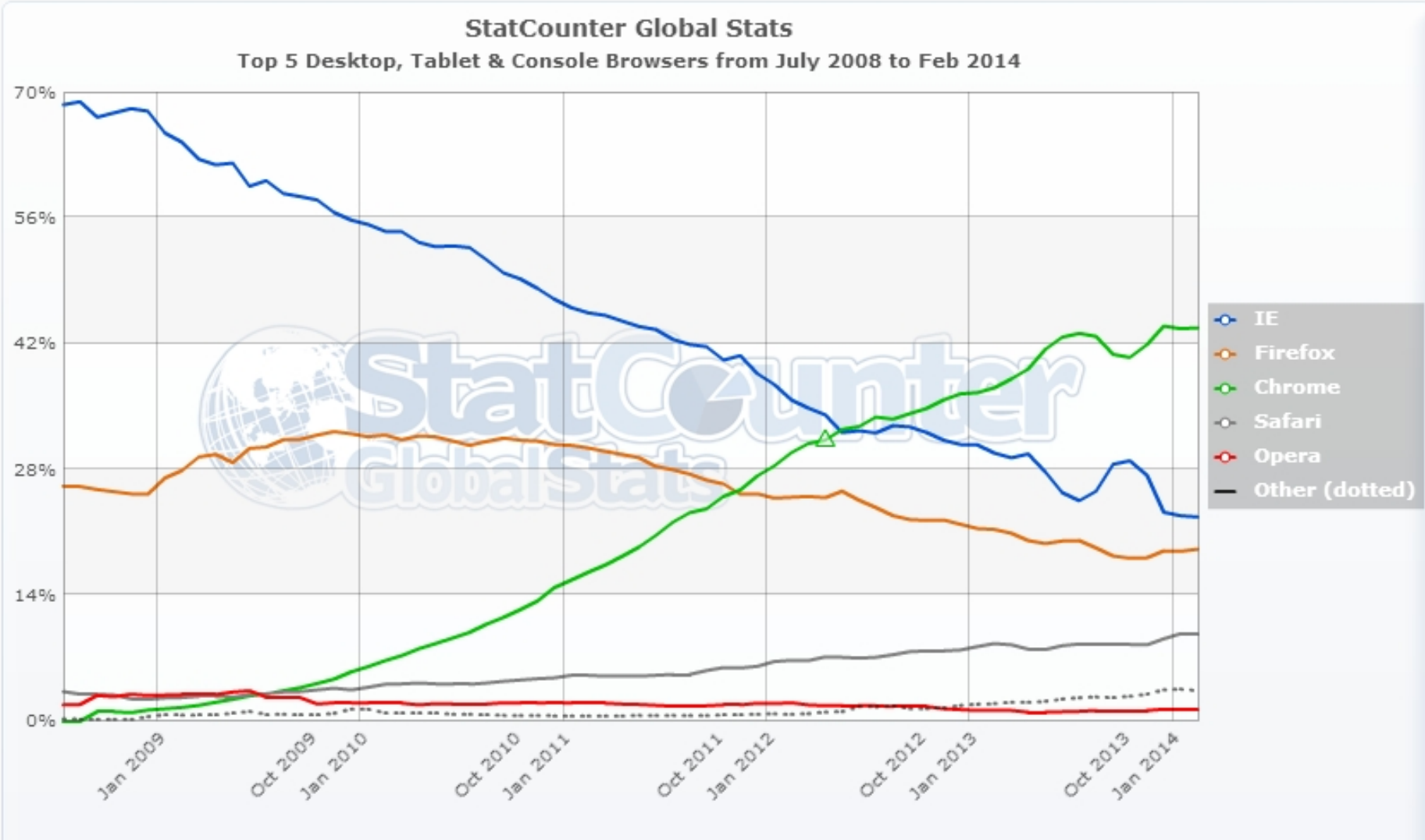
Why so many browsers

- Users are using "their" browser
- Old browsers are usually not compatible with new technologies
- If you are using a WYSIWYG tool, that doesn't mean that you'll have control of the page.
- Test your design on different browsers!
- Suggest using newer and compatible browsers to users

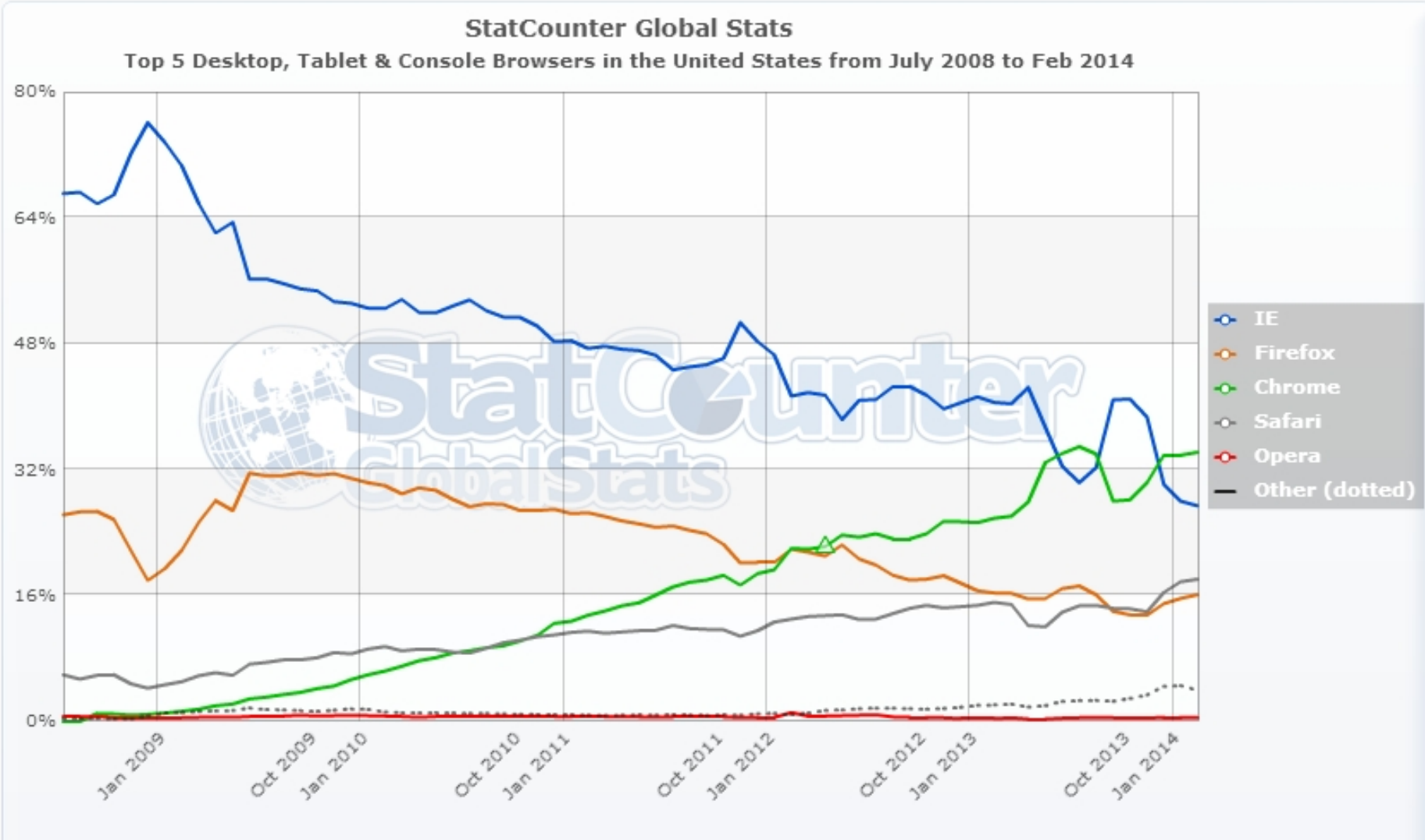
This will not help you every time, so you'll have to "adjust" to the situation...

Market share of browsers

Worldwide

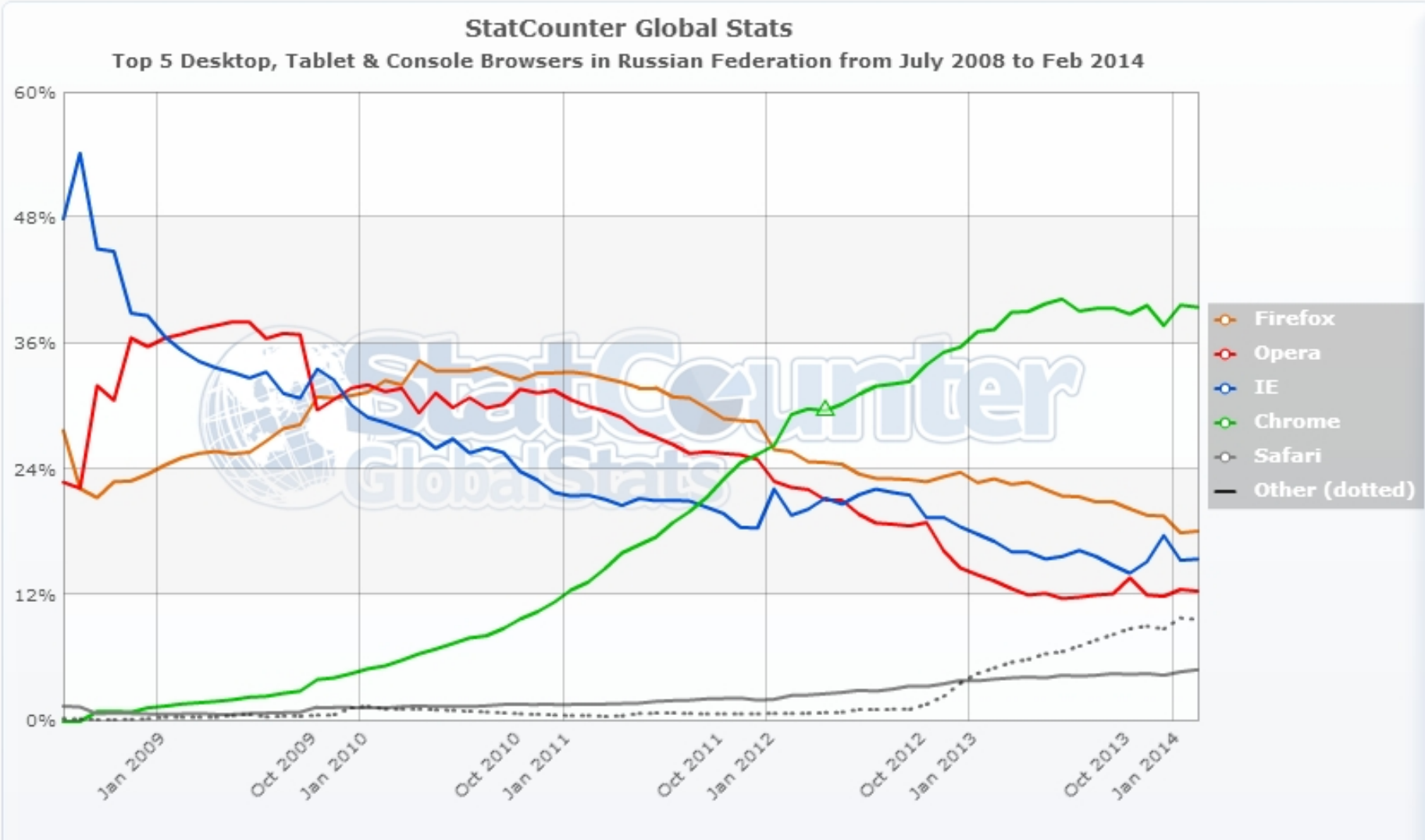


Market share of browsers
USA



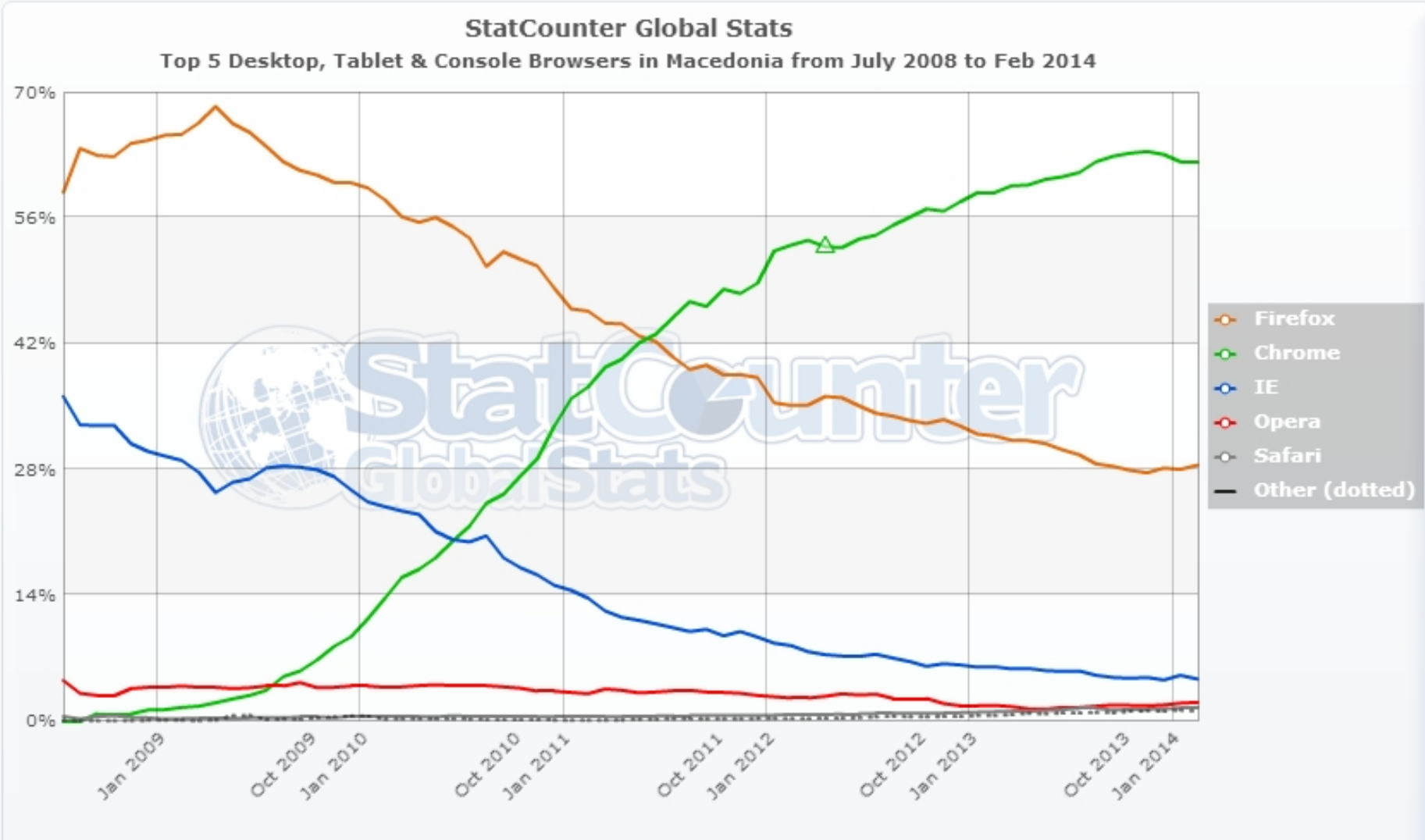
Market share of browsers

Russia



Market share of browsers

Macedonia



Browser selection

Be informed of the market share of browsers.
Be aware for local habbits. For example, Internet Explorer has around 40% share in China (mainly IE6)
All browsers are converging to the standards recomended by W3C
If you are designing "intranet" application, check for recomended browser
Be aware of the types of users (normal, advanced) and the device types that should bw working with your website (desktop, mobile, tablet...)
Usualy you'll have to target all the major browsers

How to handle different browsers?

- **DO NOT spend a lot of time tweaking! Allow for small variations!**
- Use the standards!!!
  - (X)HTML for document structure
  - CSS for presentation
- Have a good markup and spend time to design the logical order of elements.
- Do not use (X)HTML and CSS features that are not supported by every browsers.
- Learn the pitfalls of every browser :(

Document Object Model (DOM)

- DOM is a convention for representing and interacting with HTML, XHTML and XML objects in documents
- Web browsers use tools for parsing HTML documents into DOM.
- The goal of DOM is to create a structure that can be used effectively by libraries (JavaScript), for dynamical client-side changes.

Example for creating DOM from web page

```
1   <html>
2     <body>
3       <p>
4         Hello World
5       </p>
6       <div>
7         <img src="example.png" />
8       </div>
9     </body>
10  </html>
```

Listing 1: HTML Code

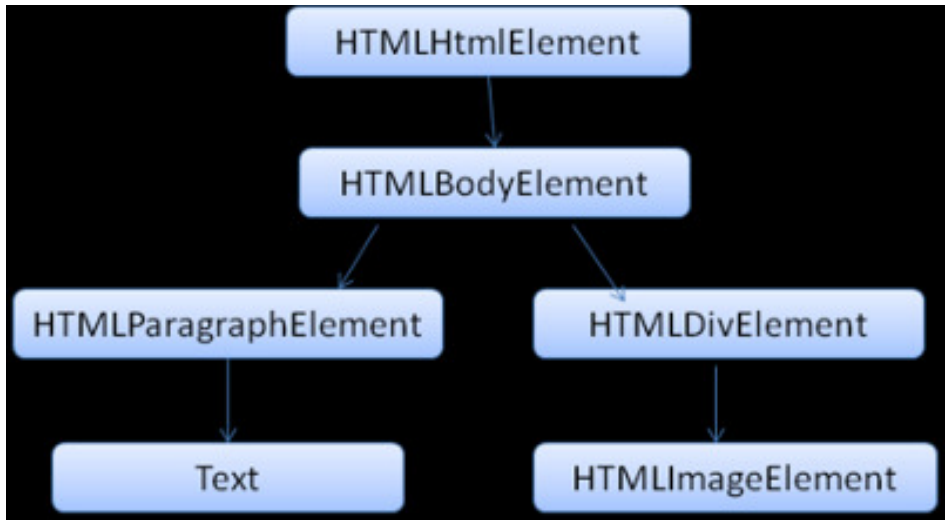
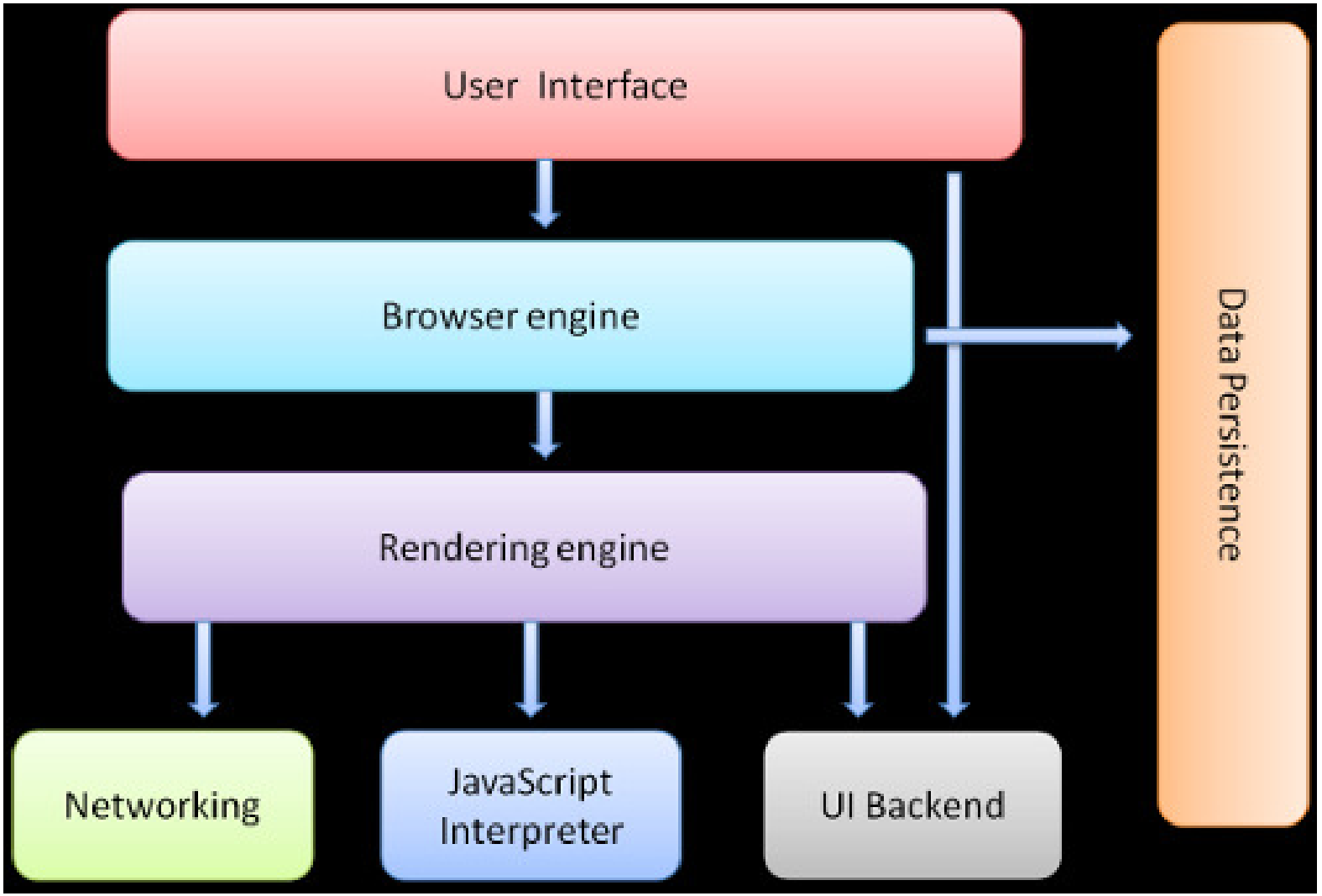


Figure : DOM Tree

Internal structure of web browsers



Interal structure of web browsers

- **User Interface** – controls that enable communication with the browser (not with the web page)
- **Browser engine** – is responsible for coordinating the action between the render engine and the UI.
- **Render Engine** – responsible for displaying the requested content.
- **Networking** – responsible for communication with the servers.
- **JavaScript Interpreter** – executing JavaScript commands.
- **Data Persistence** – responsible for storing data on primary or secondary memory.
- **UI Backend** – integration of OS specific UI components

Render Engines

All modern browsers have render engines

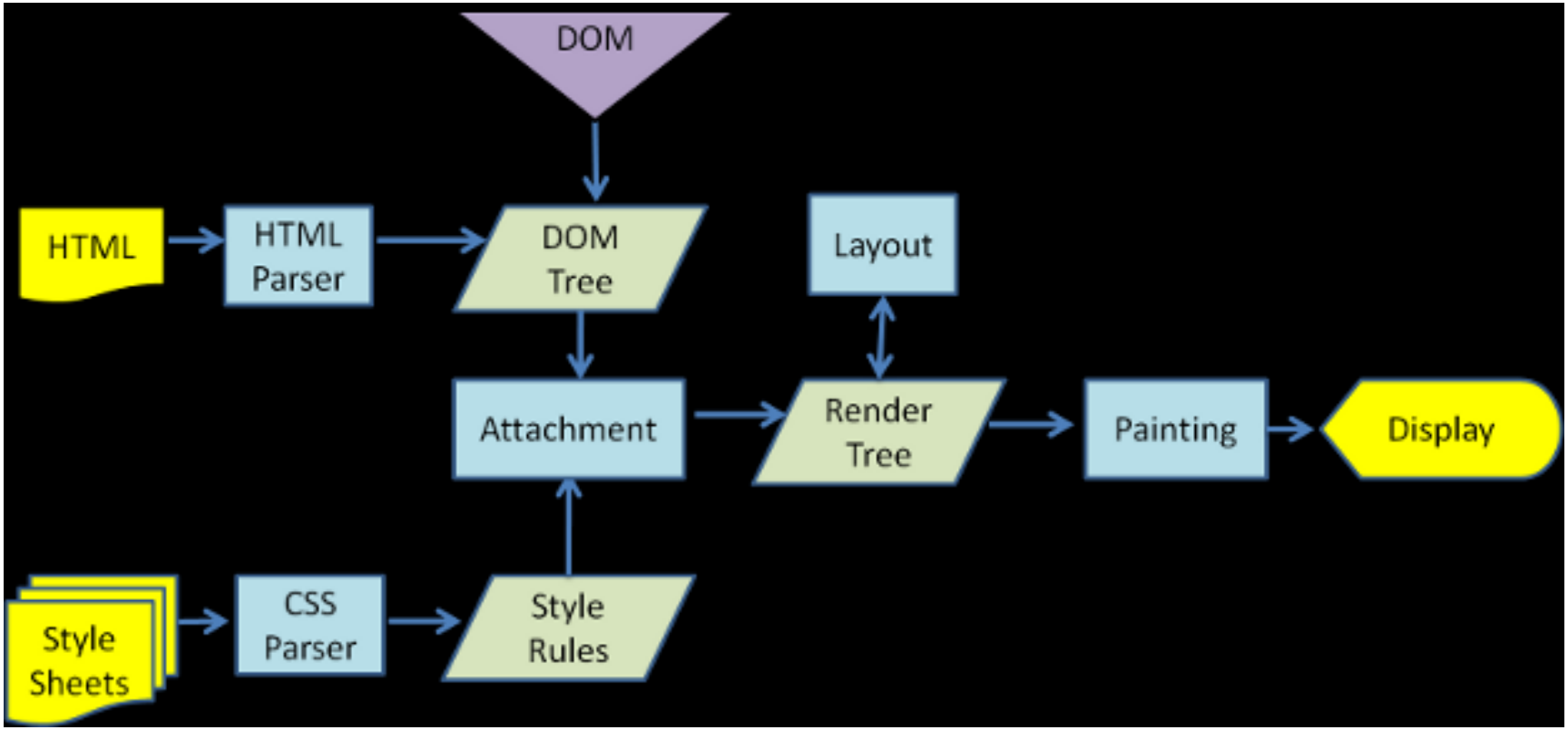
The render engine is responsible for parsing the content of the web page (HTML, CSS), creating a DOM tree and displaying it to the screen.

Common Render Engines:

- Trident (Internet Explorer)
- Gecko (Firefox)
- Webkit/Blink (Chrome, Safari, Opera 15+)
- Presto (Opera 12)

Every engine has its quirks!!!

Rendering a web page (WebKit)



Lectures, communication, news

courses.finki.ukim.mk

Questions ?