



Excellence in Software Engineering

BROWSERS. WORKING WITH DOM

THEORY & PRACTICE

Author: Viktor Pishuk

AGENDA

- Theory
 - Browsers
 - High level structure
 - Rendering engines
 - Developer's tools
 - DOM. Working with elements
 - DOM. Working with events
 - Speed up your web-page
- Practice
- Q&A



SCRUM

DEVELOPERS VS. RUGBY PLAYERS

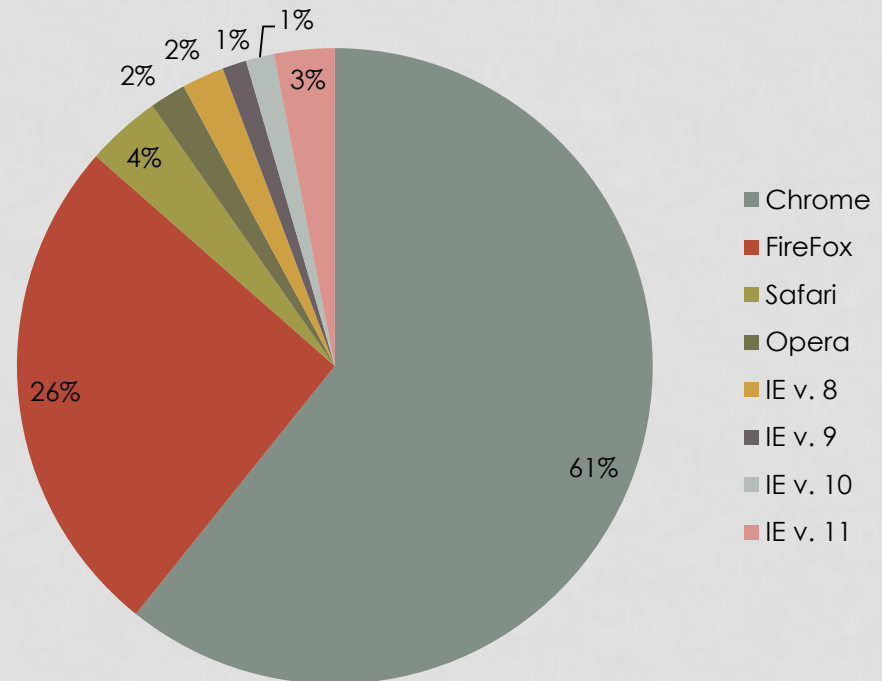
SCRUM

IT IS A PROCESS TO DELIVER EXCELLENCE IN TIME

THEORY

WE USUALLY SUPPORT:

- IE v. ≥ 8
- Chrome
- Firefox
- Safari
- Opera



RENDERING ENGINES



Gecko



Webkit



Webkit

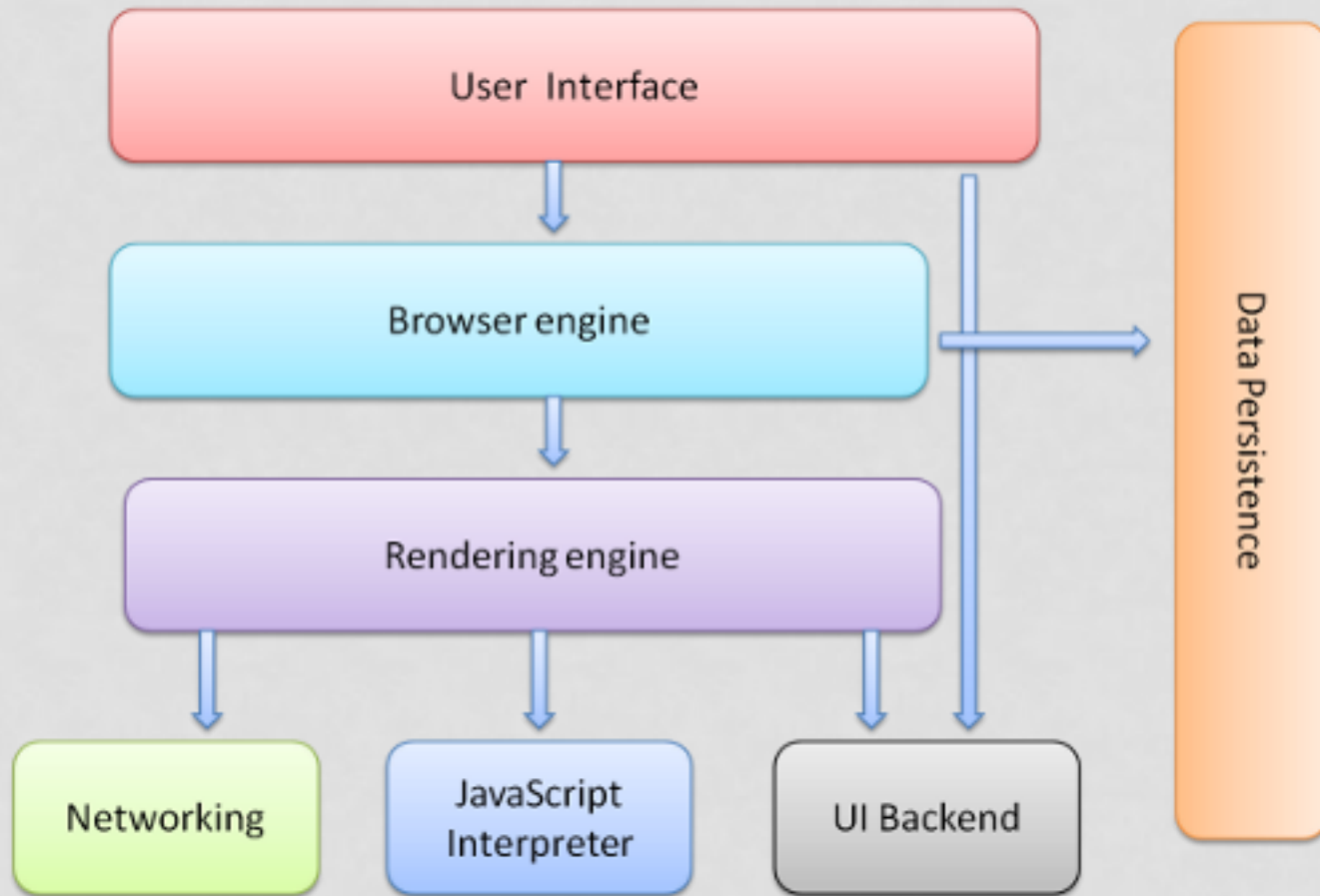


Presto



Trident

BROWSER'S HIGH LEVEL STRUCTURE



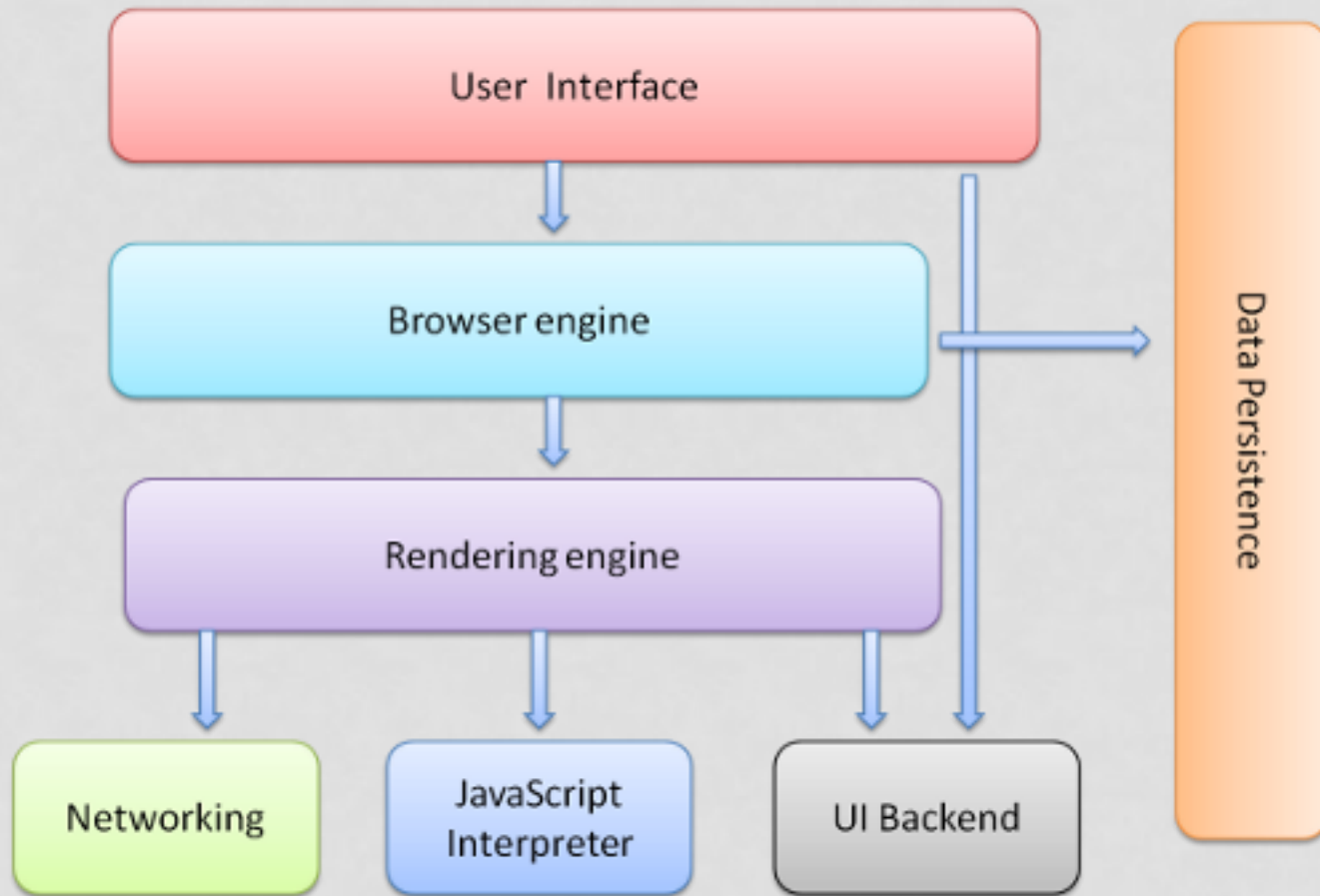
BROWSER'S HIGH LEVEL STRUCTURE



BROWSER'S HIGH LEVEL STRUCTURE



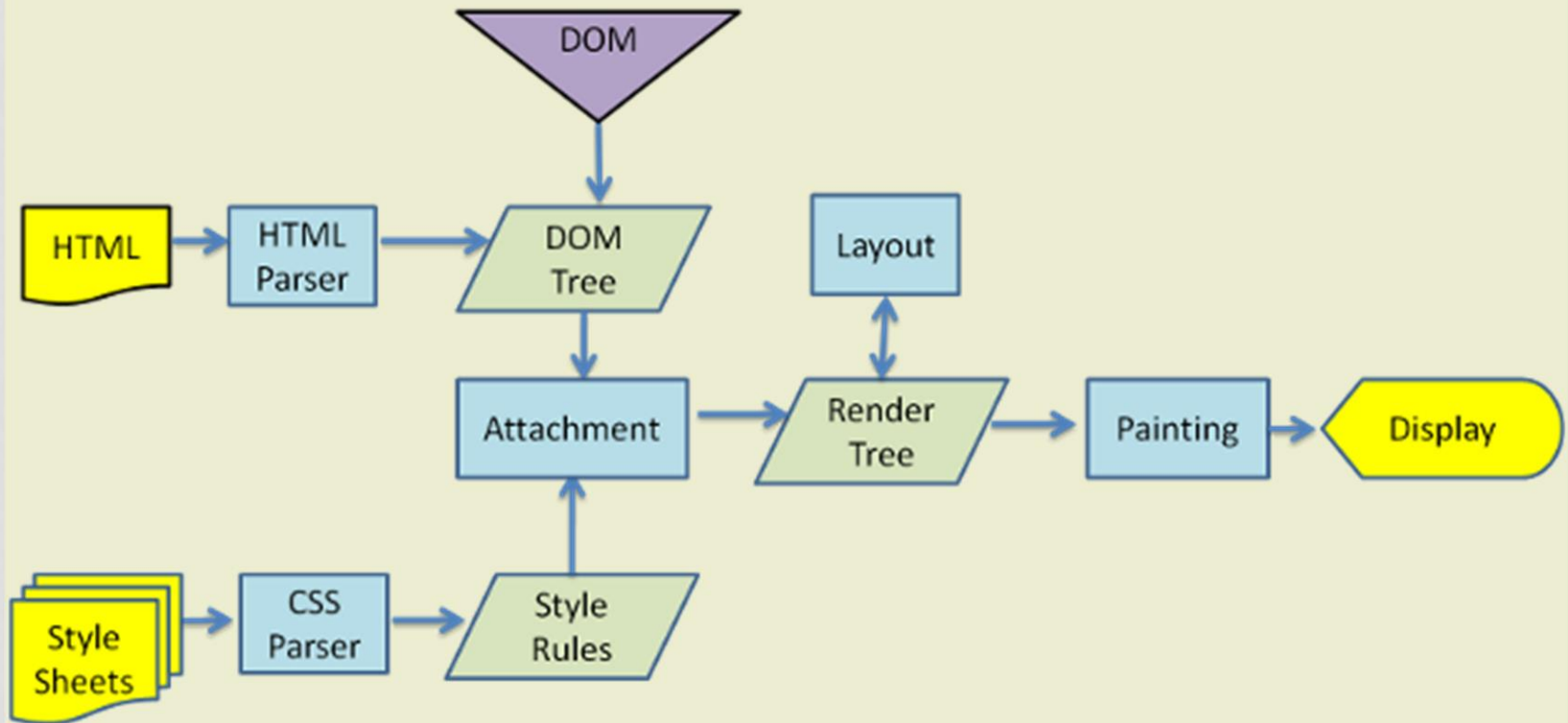
BROWSER'S HIGH LEVEL STRUCTURE



RENDERING FLOW

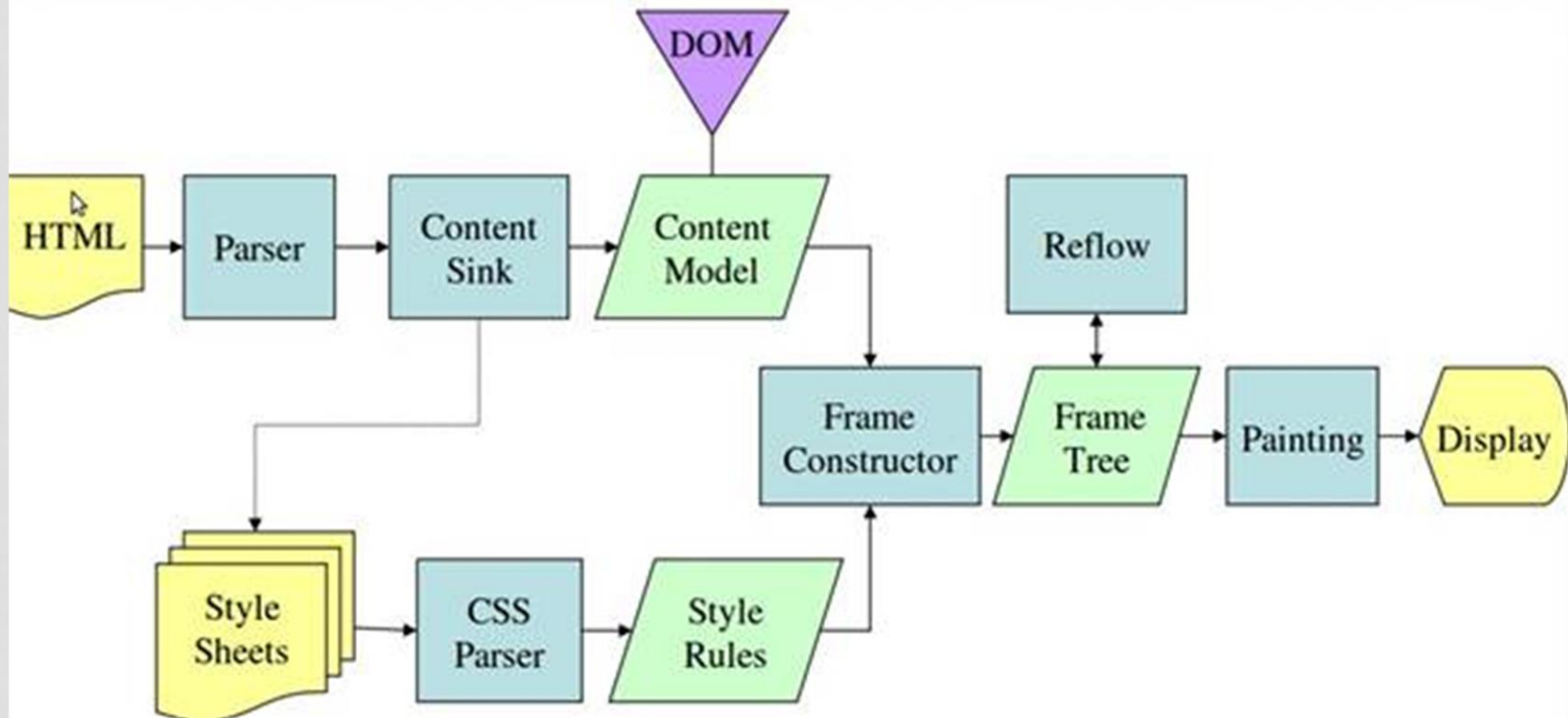


For better user experience, the rendering engine will try to display content on the screen as soon as possible.



WEBKIT RENDERING FLOW

THE CODE THAT WOULD BECOME WEBKIT BEGAN IN 1998



GECKO RENDERING FLOW

DEVELOPMENT BEGAN AT NETSCAPE IN 1997

THE ORDER

The model of the web is synchronous

SPEED UP YOUR WEB-PAGE

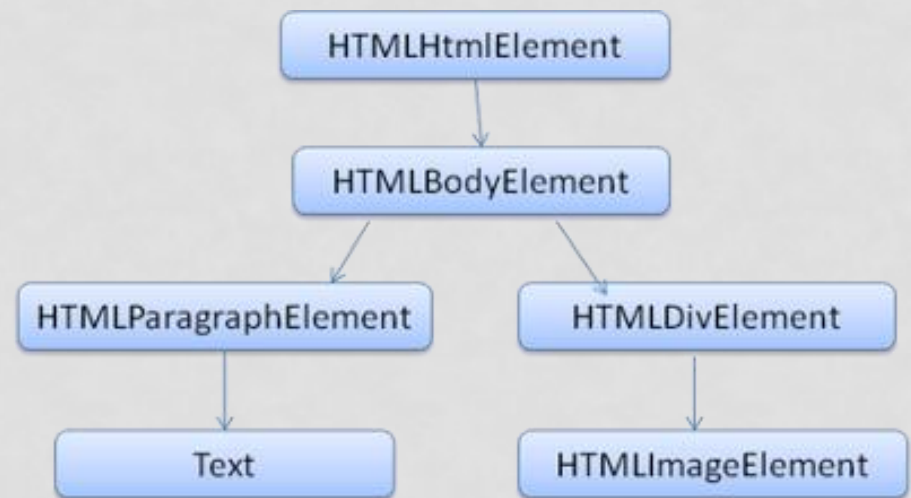
- Use following to prevent halt of rendering:
 - mark script as defer: `<script src="..." defer>;`
 - place CSS before JS;
 - DO NOT use inline JS or CSS.
- Reduce weight of JS & CSS files:
 - gzip JS and CSS files;
 - concatenate and minify JS and CSS files.
- Speed-up the load time by:
 - correctly setup cache-headers, manifest;
 - use CDN to deliver JS, CSS files faster.

DOCUMENT OBJECT MODEL (DOM)

HTML

```
<html>  
  <body>  
    <p>Hello World</p>  
    <div>  
        
    </div>  
  </body>  
</html>
```

Generated DOM



DOCUMENT VARIABLE

Root object for DOM in JavaScript is *document*

DOM INTERFACES

[illegible]

EVENTS

What events do you know?
How do you catch them?

EVENTS

Each event walks
“downstairs” and “upstairs” when arises

Phase 1: Capturing ↓

Phase 2: Bubbling ↑

REGISTER HANDLER

```
addEventListener(  
    event,  
    handler,  
    [useCapture] // false by default  
)
```

UNREGISTER HANDLER

```
removeEventListener(  
    event,  
    handler,  
    [useCapture] // false by default  
)
```

WHAT WE CAN DO WITH AN EVENT

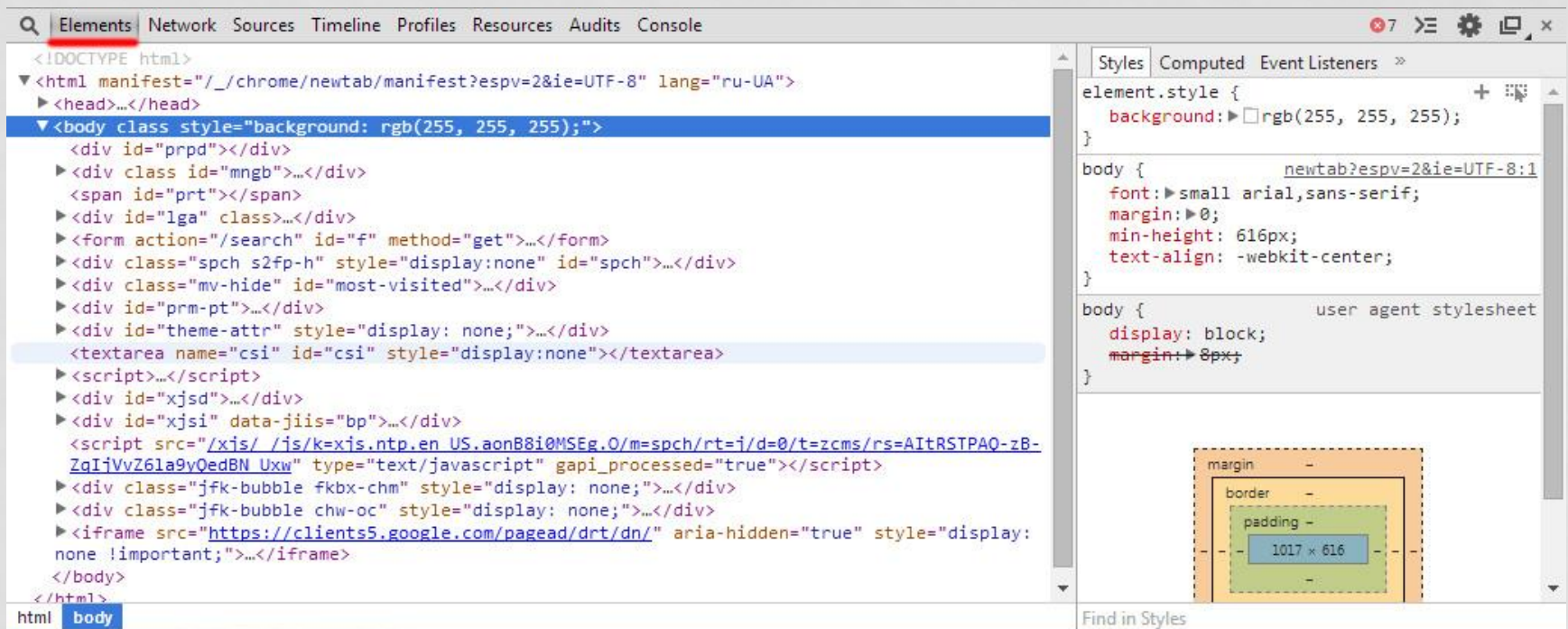
- Identify element which activate an event: target
- Identify element which listen to an event: currentTarget
- Cancel bubbling: cancelBubble to true
- Stop propagation: call e.stopPropagation()
- Prevent default handler: call e.preventDefault()

Etc.

DEVELOPER TOOLS



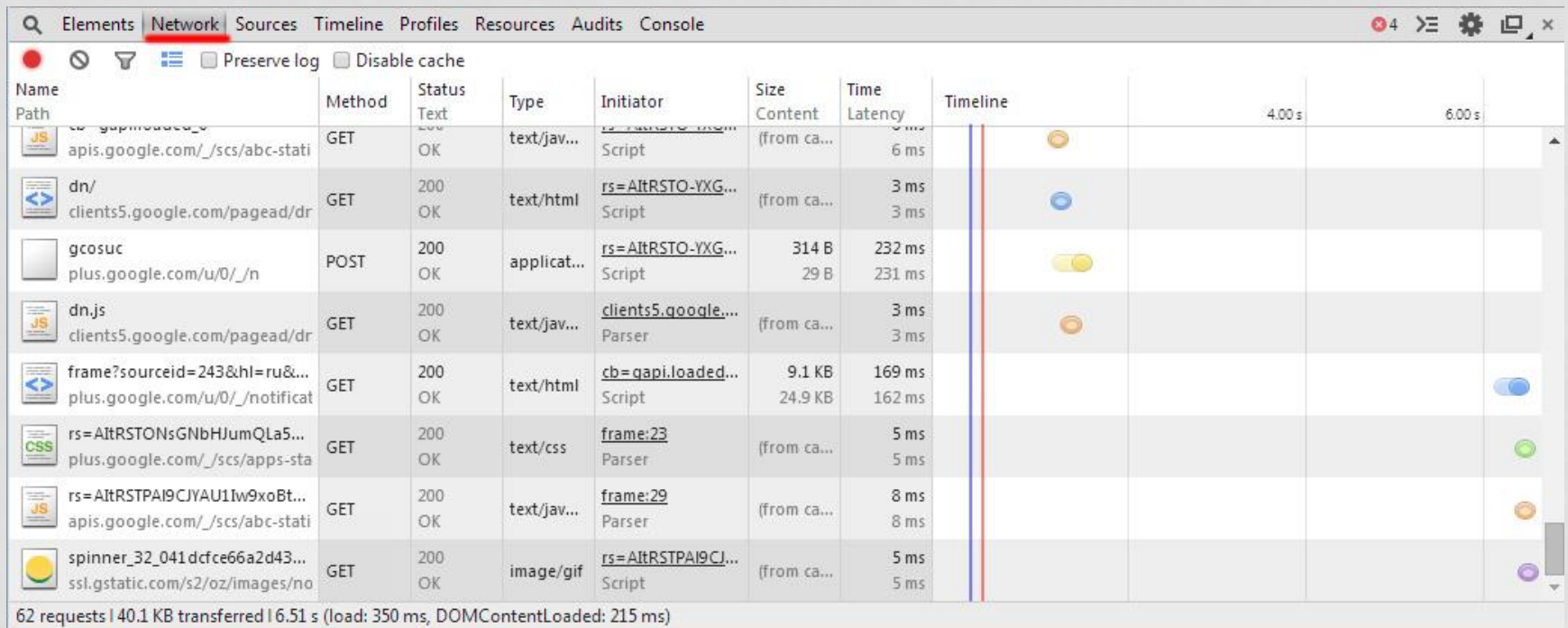
DEVELOPER TOOLS - ELEMENTS



DEVELOPER TOOLS - EMULATE



DEVELOPER TOOLS - NETWORKING

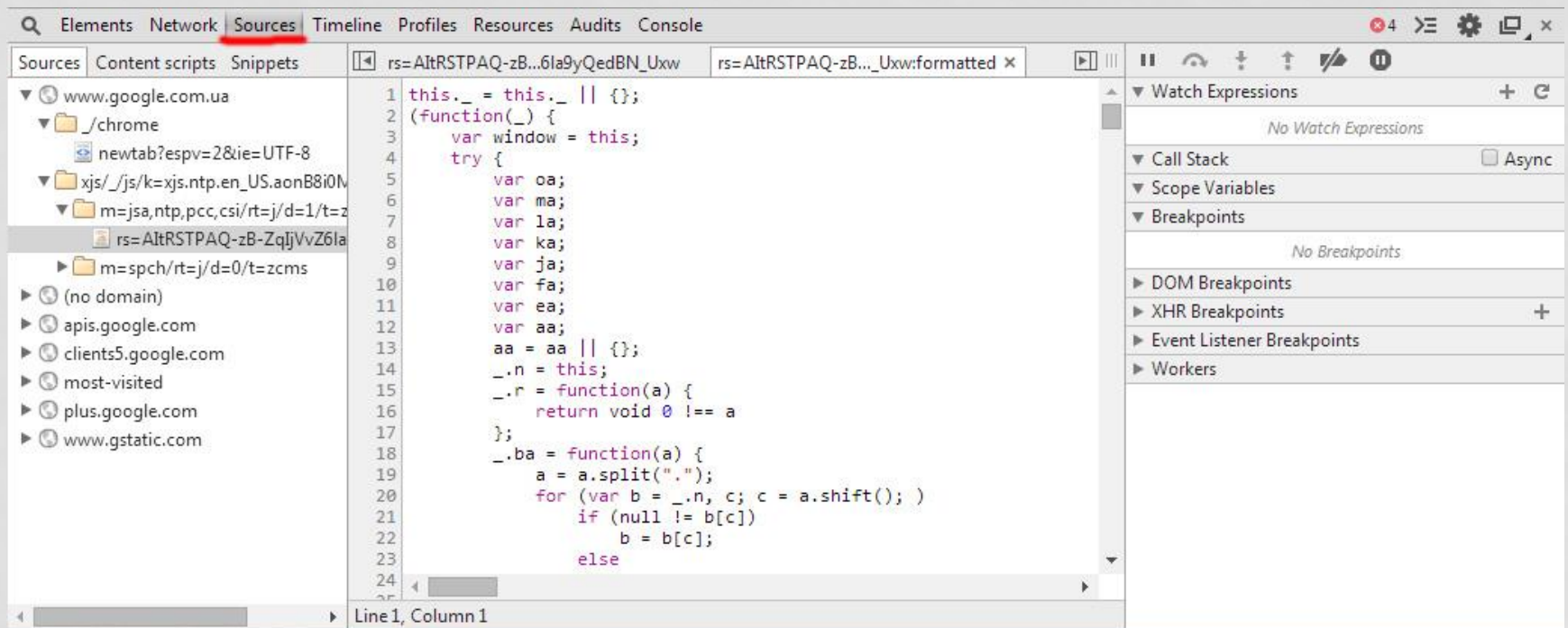


The screenshot shows the Chrome DevTools Network tab with the 'Network' panel selected. The table lists network requests with columns for Name, Path, Method, Status, Type, Initiator, Size, Time, and Latency. The 'Timeline' column shows a visual representation of the requests over time. The status bar at the bottom indicates 62 requests, 40.1 KB transferred, and a total load time of 6.51 s.

Name	Path	Method	Status	Type	Initiator	Size	Time	Latency	Timeline
apis.google.com/_/scs/abc-stati	apis.google.com/_/scs/abc-stati	GET	OK	text/jav...	Script	(from ca...	6 ms		
dn/	clients5.google.com/pagead/dr	GET	200 OK	text/html	rs=AltRSTO-YXG...	(from ca...	3 ms		
gcosuc	plus.google.com/u/0/_/n	POST	200 OK	applicat...	rs=AltRSTO-YXG...	314 B	232 ms		
dn.js	clients5.google.com/pagead/dr	GET	200 OK	text/jav...	clients5.google...	(from ca...	3 ms		
frame?sourceid=243&hl=ru&...	plus.google.com/u/0/_/notificat	GET	200 OK	text/html	cb=gapi.loaded...	9.1 KB	169 ms		
rs=AltRSTONsGNbHJumQLa5...	plus.google.com/_/scs/apps-sta	GET	200 OK	text/css	frame:23	(from ca...	5 ms		
rs=AltRSTPAI9CJYAU1Iw9xBt...	apis.google.com/_/scs/abc-stati	GET	200 OK	text/jav...	frame:29	(from ca...	8 ms		
spinner_32_041dcfce66a2d43...	ssl.gstatic.com/s2/oz/images/no	GET	200 OK	image/gif	rs=AltRSTPAI9CJ...	(from ca...	5 ms		

62 requests | 40.1 KB transferred | 6.51 s (load: 350 ms, DOMContentLoaded: 215 ms)

DEVELOPER TOOLS - SOURCES



PRACTICE

Answer: <http://jsfiddle.net/vmuha/uWcc7/>

```
document.[method | attribute]
```

Task:

Write a function to print all methods and properties of document object grouped by type into console.

Script should call console.log only twice.

You have to display strings in console, not arrays.

Tips & Tricks:

Press “Shift+Ctrl+I” to open console and use “Console” tab to write code

REMEMBER

- Functions “apply”, “call” may be used not only to change context;
- someObj.hasOwnProperty is necessary to be used when iterating through keys of any object;
- Function instanceof Object equal **TRUE**
- document object includes properties body, head

Answer: <http://jsfiddle.net/vmuha/fLje2/>

HTML Element

Task

Write a function which walks around all kids of element and do following things:

- change background color to one from the list: red, green, blue ;
- Log into console name of tag and amount of kids, use template:
Element [name of tag] has [qty of kids] children

Write a function which fills in a form

Tips & Tricks:

Use “Elements” tab to see current styles and HTML

REMEMBER

- Each HTMLElement has property childElementCount
- All kids are stored in elem.children Collection
- CSS properties like background-color accessible in camel-case e.g.: elem.style.backgroundColor
- Forms are stored in document.forms Collection and you are able to get access to any form:
 - document.forms[i]
 - document.forms.formName

Answer: <http://jsfiddle.net/vmuha/bW222/>

getElement(s)By...

Task

Write a function to show in the console:
the list of elements which are INPUT tags
the list of elements which have the same class
the element which has an id
the list of elements which has the same name
the list of elements which are INPUTs with type="button"

Tips & Tricks:

Use "Elements" tab to change HTML of current page

REMEMBER

- console.dir shows an object
- All attributes are stored in attributes property
- You are able to change any HTML code in elements tab of developer tools
- Only getElementById returns an element other methods return Collection

Answer: <http://jsfiddle.net/vmuha/x7bfq/>

Get element(s) by selector

Task

Refactor functions to use querySelector and querySelectorAll functions

Tips & Tricks:

To find element by id use “#someid”

To find element by class use “.someclass”

To find element by attribute use “[attrName=attrValue]”

REMEMBER

- querySelector returns the first element which matches selector
- querySelectorAll returns Collection of elements
- Selector to find element by:
 - id: “#someid”
 - class: “.someclass”
 - attribute: “[attrName=attrValue]”
- Selectors look like jQuery selectors but are different (see more:
<http://javascript.ru/blog/subzey/querySelectorAll-praktike>)

Answer: <http://jsfiddle.net/vmuha/3R8LJ/>

Create new DOM

Task

You have an array of 10.000 objects. Each object has few properties: name, age, position.

You have to write a function to show all objects in a grid (table). Take care of speed.

Tips & Tricks:

Use “Profiles” tab to gather necessary metrics.

REMEMBER

- Try to not manipulate with DOM directly – any manipulation with DOM is really expensive;
- When you need to generate DOM use one of the following:
 - generate a string with DOM to be inserted;
 - create documentFragment and append part of DOM to it

Events: capturing, bubbling

Task

Update a grid from previous task to show a column with delete button.

Add a handler to highlight a row by green color when user clicks on it and de-highlight on second click.

Add a handler to delete a row when user clicks on delete button.

Add a separate handler to highlight every third row by red color when user clicks on it and de-highlight on second click.

Tips & Tricks:

Sometimes you may need to catch an event in the capturing phase

REMEMBER

- It is better to catch event on parent then on each child
- There are 2 phases for an event:
 - capturing;
 - bubbling.
- To add new attribute to any element use `document.createAttribute` and `elem.setAttributeNode`
- `addEventListener` eq `attachEvent` in IE
- `removeEventListener` eq `detachEvent` in IE
- Event contains information about target node

HOMEWORK

- Use native `querySelector` instead of `jQuery` or `getElementBy` or any framework specific functionality
- Reduce time to load a web-page
- Refactor event listeners
- Do “Events: capturing, bubbling” task

USEFUL LINKS

- <http://www.w3schools.com>
- <http://javascript.ru/>
- <http://msdn.microsoft.com/en-us/library/ie/hh772384%28v=vs.85%29.aspx>
- <http://msdn.microsoft.com/en-us/library/ie/hh771866%28v=vs.85%29.aspx>
- <http://www.w3.org/DOM/DOMTR>
- https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model

CONTACTS

Author:

 Viktor Pishuk

Senior Software Engineer at “EPAM Systems Ukraine”

E-mail: Viktor_Pishuk@epam.com

Skype: victor.pishuk