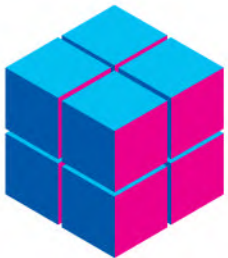




10 HOURS
FREE
VIDEO ASSETS



THE WEB DESIGN BOOK



The ultimate guide to creating stunning modern websites





THE WEB DESIGN BOOK

The world of web design changes on a daily basis. Whether it's a new language that is being invented, or a new design element, or even just a new trend or fashion. Gone are the days where coding was just about a page on the web, these days this evolved, responsive way of creating that goes as far as building games and applications for your mobile devices. In this book we aim to take you through a plethora of different ways to make your website cutting edge and brief you on the new standard of JavaScript. You will find all the tutorial files you need over on FileSilo along with over 10 hours of video tutorials. Enjoy the book!

THE WEB DESIGN BOOK

Imagine Publishing Ltd
Richmond House
33 Richmond Hill
Bournemouth
Dorset BH2 6EZ
☎ +44 (0) 1202 586200

Website: www.imagine-publishing.co.uk

Twitter: @Books_Imagine

Facebook: www.facebook.com/ImagineBookazines

Publishing Director
Aaron Asadi

Head of Design
Ross Andrews

Production Editor
Jen Neal

Senior Art Editor
Greg Whitaker

Designer
Abbi Denney

Printed by
William Gibbons, 26 Planetary Road, Willenhall, West Midlands, WV13 3XT

Distributed in the UK, Eire & the Rest of the World by:
Marketforce, Blue Fin Building, 110 Southwark Street, London, SE1 0SU
Tel 0203 148 3300 www.marketforce.co.uk

Distributed in Australia by:
Network Services (a division of Bauer Media Group), Level 21 Civic Tower, 66-68 Goulburn Street,
Sydney, New South Wales 2000, Australia Tel +61 2 8667 5288

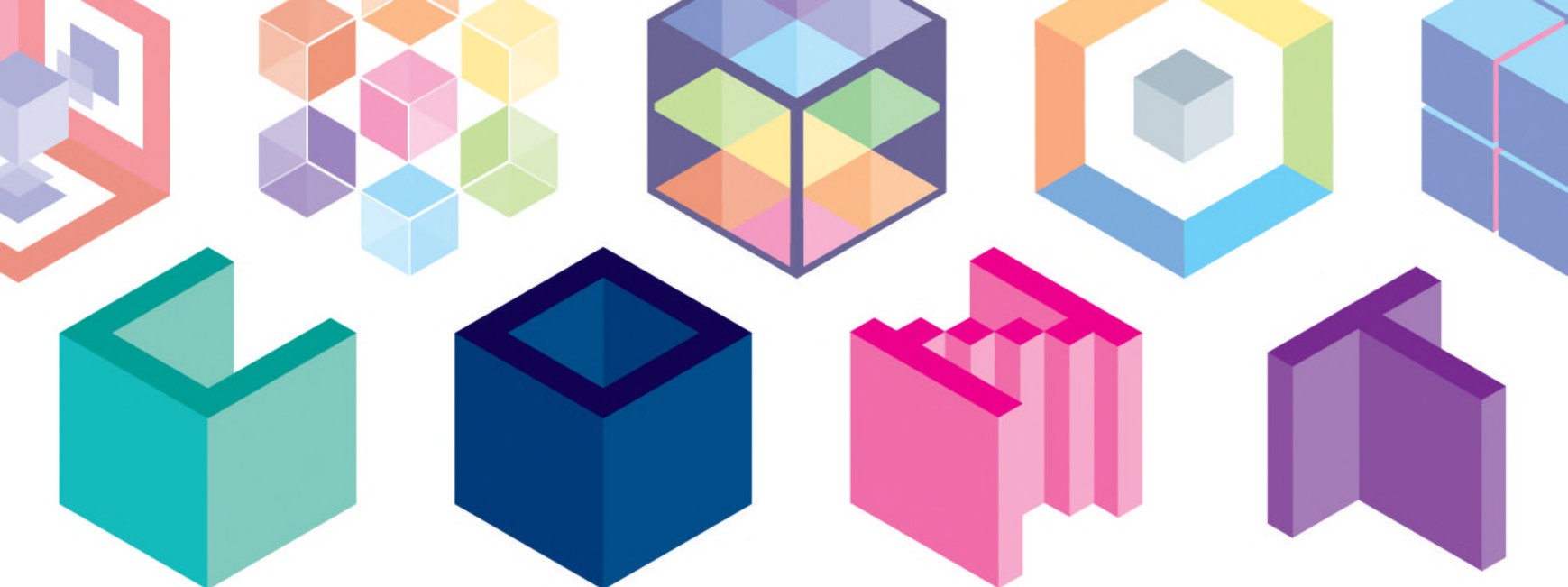
Disclaimer
The publisher cannot accept responsibility for any unsolicited material lost or damaged in the post. All text and layout is the copyright of Imagine Publishing Ltd. Nothing in this bookazine may be reproduced in whole or part without the written permission of the publisher. All copyrights are recognised and used specifically for the purpose of criticism and review. Although the bookazine has endeavoured to ensure all information is correct at time of print, prices and availability may change. This bookazine is fully independent and not affiliated in any way with the companies mentioned herein.

The Web Design Book Volume 5 © 2015 Imagine Publishing Ltd

ISBN 978-1785461101

Part of the
**web
designer**
bookazine series





Design like a pro, code like an expert

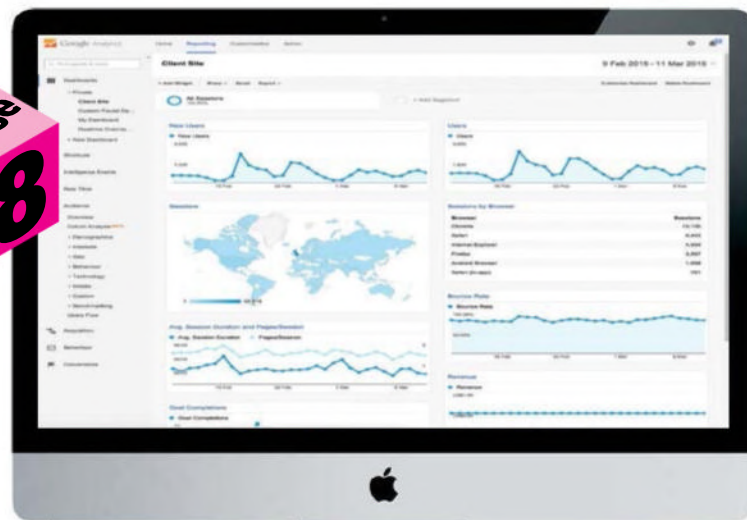
8 Create cutting edge designs

Google Analytics

16 Use Photoshop to complement your design

HTML & CSS

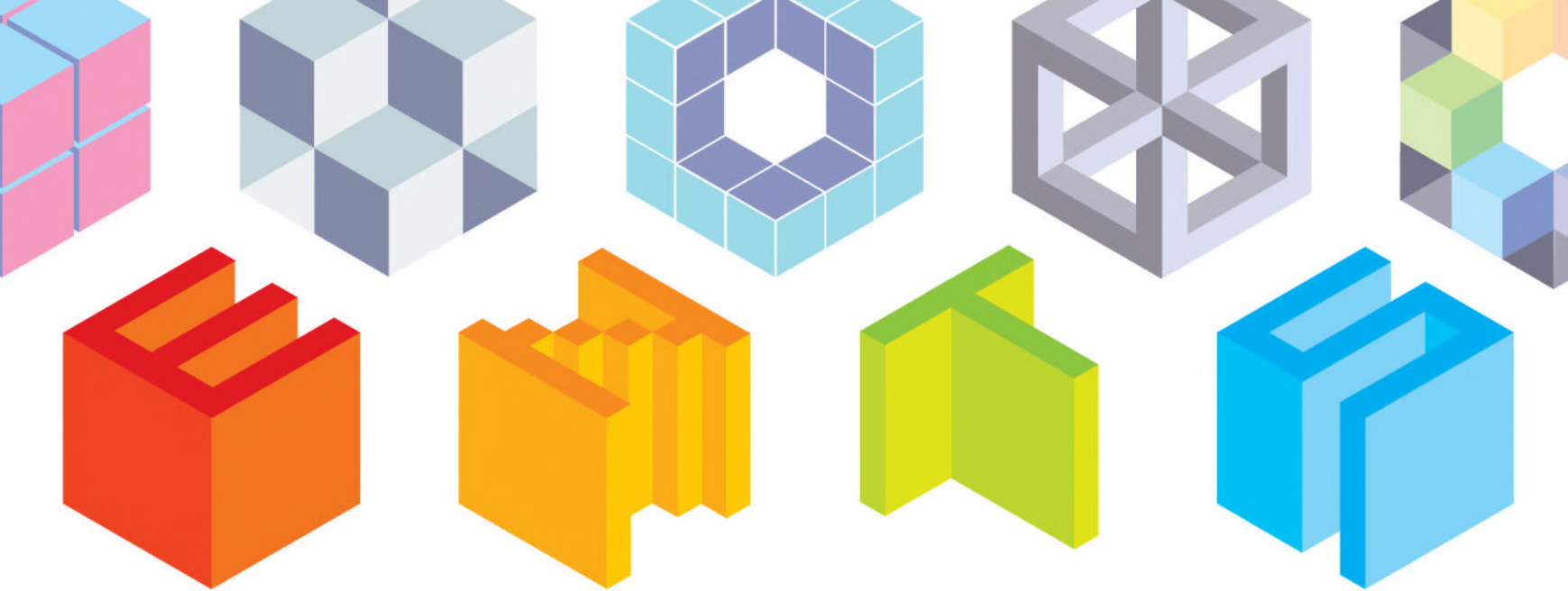
- 24 Power up CSS preprocessors
- 32 Set up your own grid system using CSS
- 36 Make pop-up modal boxes with pure CSS
- 40 Build a responsive fixed page border
- 42 Design aspect ratio based layouts
- 46 Generate sweeping animations with CSS
- 50 Make a screen shrink on scroll
- 52 Produce an animated off-screen 3D menu
- 56 Slide up titles on page load using CSS
- 58 Animate scroll-reveal split backgrounds
- 60 Animate an information card box with CSS
- 64 Construct an adaptive printable CSS design



jQuery & JavaScript

- 70 The new JavaScript standard
- 76 jQuery special effects
- 82 Make dynamic graphics with the p5.js library
- 86 Create shuffling text effects with jQuery
- 88 Design a 2D game using the Pixi engine
- 92 Sync animations to audio and video with Popcorn.js
- 96 Animate SVGs with the Vivus.js library





- 100 Code on-scroll image animations with CSS
- 102 Create animated infographics with Snap.svg
- 106 Prototype apps with Framer.js and Framer Studio

Development

- 112 20 fresh frameworks
- 118 Master advanced Angular
- 128 Construct an image gallery with CSS and AngularJS
- 132 Code validation into forms with nMessages
- 138 Create an interactive 360° scene in WebGL
- 142 Build your own API with the Hapi.js framework
- 148 Manage your project's dependencies with jspm

☞ Components will vary significantly in structure and design, even when it's throughout the same build ☞

Mobile Apps

- 154 Build apps with Facebook's React Native framework
- 158 Create a reactive web app using Meteor
- 164 Code real-time presentation applications with Socket.IO
- 168 Design and share a web app faster with React
- 172 Make a Chrome OS app with JavaScript
- 176 Build a mobile web app with NativeScript's library
- 180 Code a PhoneGap memo app with photos
- 184 Develop a reactive web app with Angular-Meteor



Code like an expert

↓ **DOWNLOAD TUTORIAL FILES**
www.filesilo.co.uk/bks-747



DESIGN

{ **LIKE A PRO** }

CODE

< **LIKE AN EXPERT** >

**Get your hands on a host of HTML5 tips, trick
and techniques to empower your frontend
and backend**

Using `<article>` for semantic components

THE `<ARTICLE>` TAG IS ONE OF THE MOST MISUNDERSTOOD ELEMENTS OF HTML5

Marking up widgets and components can be tricky to get right semantically. Components will vary significantly in structure and design, even when it's throughout the same build.

Unfortunately, this is at odds with what we actually want as developers: a consistent maintainable approach with semantic value. Well this is where the `<article>` tag comes into play.

"The article element represents a complete, or self-contained, composition in a [...] site [...] that is, in principle, independently distributable or reuseable, e.g. in syndication. This could be [...] a magazine or newspaper article, [...] an interactive widget, [...] or any other independent item of content" as defined by the HTML5 specification at w3.org/TR/html5/sections.html#the-article-element.

The specification makes it very clear that the `<article>` tag has a much wider use case than is first obvious. This makes it the perfect semantic tool for components, which we often design in modular fashions.

The HTML5 specification also details that the `<article>` tag's content model is flow content, which is the way we mark up most of the elements that are used in the body of documents. This means that the `<article>` tag becomes the semantic choice when we want to contain modular content and even functionality such as things like forms and navigation.

Of particular note is that the `<article>` tag can contain the new HTML5 elements header, footer, aside and more instances of itself - allowing for further semantic structure inside your reusable component. This provides us with semantic building blocks for complex modular UIs such as faceted search, accordion menus and tab systems - perhaps even all three at once (though this is not advised).

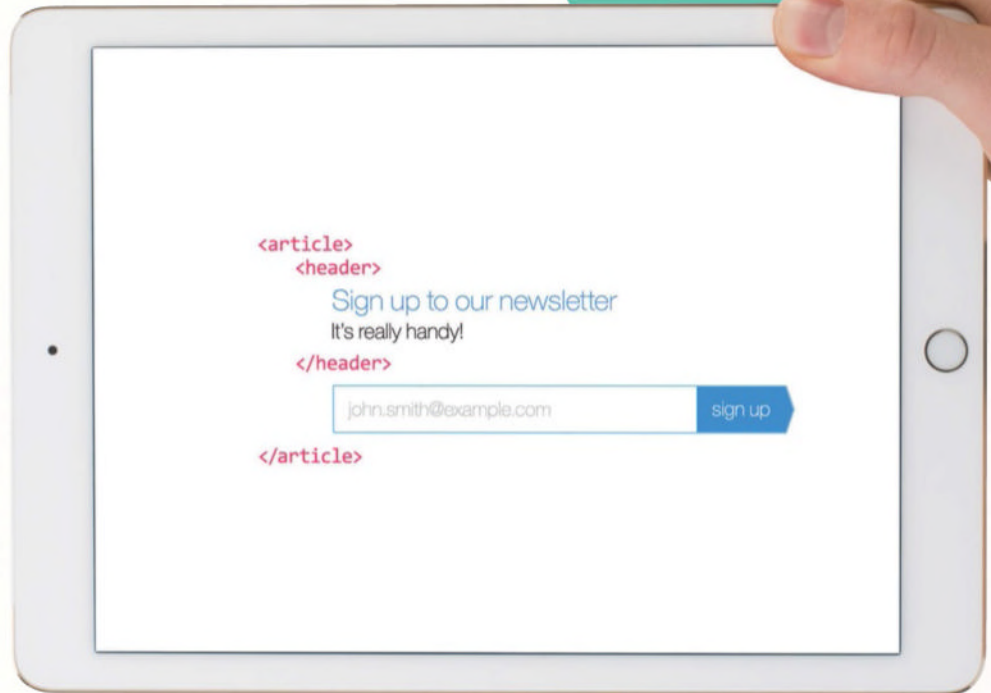
In many ways `<article>` is a 'do everything' tag. It's important to remember that the `<article>` tag brings semantic value only to components that can be isolated from the rest of the page.

It is not the right tag for supporting 'pods' or other components that only make sense in situ or are otherwise relying on surrounding content.

Likewise, the `<article>` tag is very unlikely to be the right tag in cases where the component is semirelated but also independent, for example in a pull quote. In these scenarios then, the `<aside>` tag is likely the best choice. Though it's worth considering that aside elements can contain article elements - enabling the semantic markup of any tangentially related content which also containing reusable components.

AUTOCOMPLETE TURNED OFF

Most browsers now have an autocomplete defaulted to 'on' for `<input>`. Turn this off for things like reset password fields.



“The `<article>` tag becomes the semantic choice when we want to contain modular content and even functionality”



“It’s a little detail a lot of websites miss, but form fields should only autocomplete information that is likely to be correct, lest users spend time deleting text that shouldn’t be there in the first place.”

Ralph Saunders
Front-end developer
at Redweb

Misunderstood HTML5 tags

SOME HTML5 ELEMENTS AREN'T WHAT THEY FIRST SEEM

Section

The `<section>` tag provides a semantic way to markup thematically grouped content that should be in the document outline (so it should have a heading). The `<section>` element would be perfect for marking up chapters in a book for example. We can also make use of it inside flow content, where we enable the clear separation of many sections in a lengthy article - such as a thesis or dissertation.

Footer

The `<footer>` tag is not for website footers but for footer content that relate to the nearest sectioning element (these are the four sectioning elements: section, article, aside and nav). The footer can also be used for content such as that of bylines, related documents and copyright data. It can also contain entire sections and this will then lend itself to things like appendices, indexes and licence agreements.

Header

Like the `<footer>` tag, the `<header>` tag isn't for website headers - as you would be forgiven for thinking. It's for introductory content relating to the nearest sectioning element. It can be used for larger-scale content such as a table of contents or foreword, but is more commonly used for headings and standfirst paragraphs.

Address

The `<address>` tag is not just for addresses you can post letters to. It is also contact information for the nearest `<article>` or even `<body>` when it's applicable to the document as a whole. While this may be a postal address sometimes, it's more likely an email address, phone number or link to the author's page.

Nav

The `<nav>` tag can be used for primary navigation, but it should also be used to markup any navigation on the page. Secondary navigation, even when internal to the page, should be marked up with `<nav>` - like jump links. If the nav contents represent a list of items they should be marked up with a list as well.

HTML5 CHEAT SHEET
There are a lot of tags in HTML5, some popular and some not so well known. A quick reference to a host of HTML5 tags is just the job. Get your hands on one at websitesetup.org/html5-cheat-sheet.



ARIA landmarks

USE ARIA LANDMARKS TO IDENTIFY REGIONS OF A PAGE

ARIA stands for Accessible Rich Internet Applications and is a set of attributes for making markup more accessible. The basic premise is that these attributes can help describe the components on a page for screen reader applications. A subset of these attributes are called landmark roles, and they indicate navigational landmarks like in this code example:

```
<article role="main">
  <header>
    <h1>Awh, kitteh</h1>
    <p>We've just named our new cat
    Jimmy!</p>
  </header>
</article>
```

Here we're declaring this section of the page the 'main' content of the document - it contains the content you came to the page for. There are a few more landmarks that should go into any web project.

'Banner' declares that the region is mostly site oriented as opposed to page oriented. This can be applied to site headers where the logo, primary navigation and search are site specific.

'Complementary' declares a region that's outside the main content but complements it and would remain meaningful in isolation. Most sidebars fit this criteria.

'Navigation' declares the region contains navigational elements for navigating the page or related pages. If you're using `<nav>` you should be using `role="navigation"`.

Optimise page structure for Time To First Paint

OPTIMISE FOR TTFP TO GET MORE PAGE VIEWS AND IMPROVED CONVERSIONS

TTFP is the metric for measuring how fast the browser was able to paint the page. We

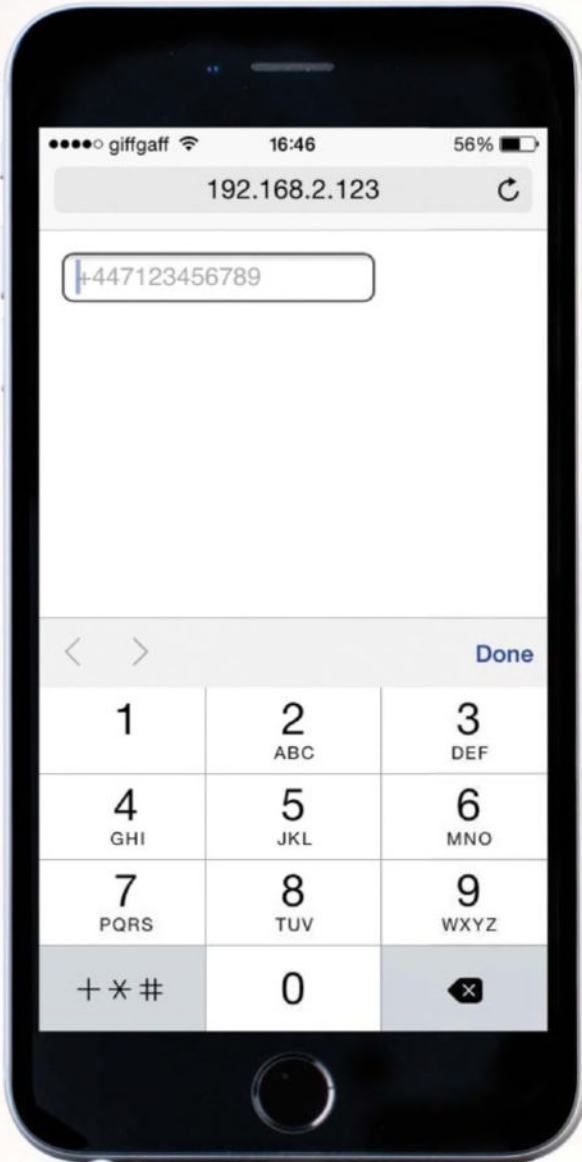
want to minimise the time a user spends looking at a white screen by optimising assets and the order in which they're loaded.

When visiting a page the browser must download the HTML of that page before it fetches the CSS and JavaScript assets linked within it. There are a couple of things we can do to ensure only the minimum number of assets are fetched before the browser paints. Most people know that JavaScript should go at the bottom of a page just before the `</body>` tag. What's less known is

that if your JavaScript isn't interacting with the CSSOM you can include an `async` attribute to turn it into a nonblocking request, meaning it won't hinder TTFP. `<script src="app.js" async></script>` We can also use the media attribute in links to CSS assets to make them nonblocking.

```
<link href="print.css" rel="stylesheet"
media="print" />
<link href="desktop.css" rel="stylesheet"
media="(min-width: 1024px)" />
```

Desktop.css won't delay TTFP for devices that don't meet the requirements of the media query, it gets loaded later.



HTML5 input types for better keyboards

BRINGING NATIVE INPUT VALIDATION, AND UX GOODIES TO THE TABLE

Mobile devices often have small keyboards where data entry is made all the harder. We can ease our users' pain by using the new input types that come with HTML5, which can change a keyboard layout intuitively on some mobile platforms out there.

While using these types correctly won't fix a bad form, the different layouts go a long way in making forms less stressful to use on small devices. Some of the more common input types you can include in your forms straightaway are below:

`<input type="tel" name="telephone">`

Tel inputs will bring up a numeric keyboard. No more pesky small keys when entering phone numbers. No format validation is enforced with this type.

`<input type="number" name="amount of pets">`

Number inputs will also bring up a numeric keyboard. Only numbers can be put into this input.

`<input type="email" name="email">`

Email inputs will bring up a keyboard with an @ sign.

Browsers will do email address validation on this input type.

`<input type="url" name="website">`

Url inputs will bring . / and .com buttons to the interface. The browser will validate that this looks like a URL.

`<input type="date" name="date of visit">`

Date inputs will bring up a date picker on modern desktop browsers and a scrolling interface on most mobile devices.

“Different layouts go a long way in making forms less stressful”

Use rel="prefetch" to enhance browsing experience

PREFETCH CAN SPEED UP BROWSING EFFECTIVELY

HTML5 prefetching uses the rel attribute on <link> tags and looks like this:

```
<link rel="prefetch" href="/Page2.html" />
```

Page2.html will be fetched by the browser when nothing else is happening and it's fetched before the user decides to go there. Server response time is often the largest delay in fetching new pages and other assets will be cached, the savings here are substantial.

URL prefetching is a double-edged sword. We're downloading a resource that the user may not ever navigate to and on a mobile network it'll count against your user's data allowance. With this in mind, we should only prefetch resources we're pretty sure the user will visit. There are some common resources that are always worth prefetching:

1. The next page in paginated search results
2. The next page in sequential multipage forms
3. Pages with only one major call to action that the user will likely visit eventually
4. Common images on most pages, like spritesheets.

Along with standard prefetching there's also DNS prefetching which works in a similar fashion.



Darren Hickling

Web engineer at Redweb
 “HTML5 introduced new input types, but also new attributes for them. Alongside pattern, these new input types have proven very useful in recent projects. In supported browsers, they can also remove the need for custom JavaScript validation and are enforced by the browser so they cannot be ignored.”



Phil Heywood

Creative director at Redweb
 “Web pages using dynamic content and user interface components used to cause problems for people with certain accessibility needs. HTML5, in conjunction with WAI-ARIA, makes it possible for everybody to use rich Internet applications, even if they rely on screen readers and other assistive technologies.”



Ryan Kerry

Global head of development at Lush
 “HTML5 introduced the contentEditable attribute, which enables users the ability to alter the contents of a HTML page. This has proven to be extremely useful when required to perform rapid in-browser prototyping of web pages that require real-time updates to the CSS output for clients to see.”



Jonathan Ginn

Developer evangelist at We Are Base
 “One of my favourite underutilised new HTML elements is the datalist. It's great for letting someone perform a search while offering find-as-you-type options, all in HTML5 with no JavaScript. It's not fully supported yet but like all HTML5 elements, it has graceful fallbacks for older browsers.”

Back-end APIs for apps

USE HTML5 TO MAKE APPS WITHOUT NATIVE PLUGINS

WebRTC

Don't be fooled, WebRTC has nothing to do with real-time clocks. Instead, brace yourself for a standards-based way to perform voice and video calls using nothing but a web browser. When sufficient traction has been gained, Web Real Time Communication might swipe away Skype, MSN Messenger and other proprietary protocols. Sadly, its adoption has been slow so far - it is limited to Chrome, Opera and Firefox only.

WebWorkers

JavaScript and multithreading are like cat and water: when the language was first designed, no one thought of multicore processors.

WebWorkers seek to remedy this design deficiency by introducing a special and convenient way to run JavaScript logic in a separate thread. A message-passing protocol is then used for data exchange between the worker and the main app.

Shadow DOM

Even though Web Components have not been formalised as of this writing, the shadow DOM feature already deserves attention. It permits the isolation of subcomponents from the rest of the page. When stored in a ShadowRoot instance, your widget's elements are shielded from well-meaning but badly written enumeration routines seeking to wreak havoc on their finely tuned composition.

Typed arrays

If JavaScript wants to become a full-featured programming language, a way must be found to enable interaction with native libraries. Sadly, its untyped memory model makes the management of memory buffers difficult. In essence, a typed array is little more than a broker between a slab of memory and JavaScript's native types. You can use them to create arrays of floats or integers though, and these can then be passed onto any native functions.

File API

HTML5 introduces a plethora of storage systems. Sadly, sometimes there is just no alternative to a good old file. The File API permits you to access pesky little and chunky large files sitting in the file system of your client's workstation. It comes complete with a set of widgets to make selecting the files easy.

Where are we?

THE GEOLOCATION API HELPS YOU FIGURE OUT THE USER'S LOCATION



1. Where is our user?

Even though the HTML5 Geolocation API has seen some impressive implementations, some browsers still don't implement the library correctly. The first step to a working GIS application involves the querying of the presence of the geolocation object. If found, the location getter methods can then be invoked. GetCurrentPosition takes one, two or three parameters: one method which will be called on success, an optional second one which will be called on failure and an optional JSON object with further parameters to set up accuracy and data source selection.

2. Handling all errors!

Getting a user's location is all but foolproof. The optional error handling method receives an object containing four different error codes, eg PERMISSION_DENIED is returned if the user denies the location request shown as the website tries to access the Geolocation API for the first time. Note that some browsers don't return an error if the user refuses to divulge to your application. On such browsers, the success callback simply doesn't get invoked - one way to handle this behaviour is with setTimeout. It can spin a checker to inspect whether success and/or failure have invoked.

3. Parsing user data

If the locationeering process was successful, you are provided with a position object. It contains the set of fields shown in the following table:

Field	Data
accuracy	Accuracy of Lat and Lon data
altitude	Altitude from mean sea level
altitudeAccuracy	Accuracy of altitude
heading	Heading, degrees from north
latitude	Latitude
longitude	Longitude
speed	Movement speed
timestamp	Age of data

4. Caveat emptor

Keep in mind that the actual implementations differ on legacy browsers. For example, Internet Explorer did not support the official Geolocation API until IE9. Fortunately, an intermediary broker layer has been developed. It acts as a relay station between your code and a variety of proprietary implementations. Further information on it can be discovered by visiting the website of the project (github.com/estebanav/javascript-mobile-desktop-geolocation). Another source of grief involves the accuracy of the data: most desktops are located via their IP address. This can lead to significant accuracy issues.

10 HTML5 tips

MAKE THE MOST OF YOUR BACKEND WITH THESE TRICKS

HTML Imports are dead

During the specification of Web Components, Mozilla came up with an include directive for HTML files:

```
<link rel="import" href="myfile.html">
```

Sadly, they never managed to pick up any form of significant traction. Since Firefox dropped support for them some time ago, it's better to leave them alone.

Force some input

HTML5 saves validation effort by informing the browser that some fields must be filled:

```
Name: <input type="text"
name="mandatoryname" required>
```

By default, most browsers will display an exclamation mark in the field as long as it is empty.

Force a pattern

For some types of data, a predefined format is very helpful for us as it can be enforced in order for us to weed out any invalid input from our code. Take a look at our snippet below because it shows how HTML5 can be harnessed in this instance to ensure that our users don't end up accidentally inputting non-IPv4-formatted addresses:

```
<input type="text" pattern="\
d{1,3}\.\d{1,3}\.\
d{1,3}\.\d{1,3}">
```

Annotate data

HTML always found itself in a pressure field between verbosity and compactness: adding new tags makes the language richer, but increases complexity.

Microdata solves this by specifying an external format for metadata. The markup is then enriched with tags referencing the metadata stylesheet:

```
<section itemscope
itemtype="http://data-
vocabulary.org/Person">
<h1 itemprop="name">Mark
Pilgrim</h1>
```

Specify your own

All kinds of actors have created a large selection of metadata stylesheets, which can be embedded into your designs easily. But of course, a custom stylesheet is needed in some sticky cases. More information (as well as a large range of information on the microdata data model, markups and snippets) and these can be found at the website of Mark Pilgrim, which is available at diveintohtml5.info/extensibility.html.

Using gauges

Digital aeronautical systems have always suffered from a weakness of the human brain: textual information is more difficult to parse. Fortunately, Gauges can be created on HTML5-capable browsers by using the following, widely supported tag:

```
<meter value="6" min="0" max="10">6 of 10</
meter><br>
```

<Progress> and <meter>

Even though there is nothing to stop you from using a Gauge as a progress bar, HTML5 provides a dedicated option. Its use is demonstrated in the following snippet - keep in mind, that some versions of Internet Explorer don't support the <progress> and <meter> tags:

```
<progress value="66" max="100">
</progress>
```

Check your spelling

Typos can and will happen in the best of houses. Some browsers include a spellchecking engine, which can be set loose on all kinds of input fields. Enabling the spellchecker is as easy as setting the following extra attribute: `spellcheck="true"`

Stay compatible

Not every browser supports every part of each and every HTML5 specification. **CanIuse.com** is the semiofficial go-to point for developers wondering whether their feature of choice will impede compatibility with the common and not so common browsers too.

Don't get hacked

Keep in mind that any and all HTML5 components are pure client-side stuff. The availability of open source browsers has enabled hackers to create an 'evil browser' which does not comply to standards - skipping server-side validation is a sure-fire way to ask for pain.

5 Tools

FIVE ADDITIONAL FEATURES TO MAXIMISE YOUR WORKFLOW

Keep them offline

Caching has been around since the glory days of primitive web browsers. HTML5's application cache makes the cache engine addressable. Your code can inform the browser about elements which should be cached. The most common use case involves the creation of web apps, which keep running if deprived of their internet connection. In addition to that, the API can also be used to cache images.

Web storage bingo

In many cases, persistable data consists of but a few small tuples. In this case, the Web Storage API is the tool of choice. It provides a simple KV-Store, of which values can be persisted across sessions or across invocations of the web application. Beware, however, that most browsers tend to put tight constraints on the maximum size of the storage partition.

Database ahoy!

Sometimes, databases are needed to handle large and/or more complex data scenarios. The IndexedDB API provide an object oriented NoSQL database which can be used even in offline mode. As with web storage, be aware that the amount of space available to your application is likely to be limited in order to prevent local denial of service attacks by malicious or badly-designed websites.

Modify the back stack

Windows Phone 7 acquainted users with the back button. After having used one of Microsoft's handsets, users expect the back-button to work coherently. The History API permits you to manipulate the history of the browser. You can modify the address shown in the address bar, but you can also insert steps into the back queue just in case the user ever feels like backtracking through a process.

Even more technology

The web is evolving at an amazing speed. If you want to know more about new technologies, take a look at Mozilla's WebAPIs (developer.mozilla.org/en-US/docs/WebAPI) as it's very interesting to read. The company is in the unique position of having to maintain a completely web-based operating system. This unveils interesting problems normally not faced by browser vendors. Plus, Mozilla's drafts have been accepted as standard frequently.



Go 3D with WebGL

TAKE THE HARD WORK OUT OF CREATING 3D BY FIRING UP THREE.JS

Venturing into the third dimension has always fascinated programmers and designers. Even though 3D engines have been implemented successfully using nothing but JavaScript, their performance was incredibly low due to the lack of hardware acceleration. Fortunately, a technology called WebGL seeks to ameliorate this issue. On a supported browser, WebGL code is passed directly to the GPU and is run at speeds almost comparable to code written in DirectX and/or OpenGL.

Sadly, handling WebGL on its own is tremendously difficult: if you ever bothered yourself with creating OpenGL or DirectX rendering code, you are very well aware of the endless amount of pain hidden in this particular field of computer science.

Fortunately a nice wrapper framework called three.js is available. It acts in a fashion similar to XNA, providing

developers with a user-friendly frontend and taking care of the ugly and gory mathematical and computer scientific details transparently in the background.

While running WebGL on the desktop is by and large not an issue, support in the mobile space is spotty at best. For example Firefox OS was unable to accelerate WebGL scenes until the arrival of Firefox OS 1.2 - developers and users stuck on older versions of the OS need to live with agonisingly slow performance. In addition to that, the very popular Opera Mini browser also has its fair share of issues with WebGL. WebGL is tremendously cool, as can be seen by looking at the rendering code and the screenshot shown above. Sadly, its compatibility in the mobile space is limited - but since Adobe has abandoned Flash for mobile when it met resistance from the late Steve Jobs, developers have little choice.

Code library

For the full code in this library, check out our FileSilo

```
var renderer = new THREE.WebGLRenderer({ alpha: 1 });
renderer.setClearColor(0x0055ff,1);
renderer.setSize( window.innerWidth, window.innerHeight );
var loader = new THREE.JSONLoader();
loader.load( "Monkey.json", function( geometry 2 )
{
  geometry.computeBoundingBox();
  mesh = new THREE.Mesh( geometry, new THREE.MeshNormalMaterial() );
  mesh.scale.set( 10, 10, 10 );
  mesh.position.y = 0;
  mesh.position.x = 0;
  scene.add( mesh );
  var pointLight = new THREE.PointLight(0xFFFFFF);
  ...
  for(i=0;i<9;i++)
  {
    var geometry = new THREE.TorusGeometry( 10, 3, 16, 100 );
    geometry.computeBoundingBox();
    var material = new THREE.MeshLambertMaterial( { color: 0xff00ff } );
    var torusMesh = new THREE.Mesh( geometry, material );
    torusMesh.position.x=2+getRandomNumber(30);
    torusMesh.position.y=2+getRandomNumber(30);
    torusMesh.position.z=2+getRandomNumber(30);
    torusMesh.scale.set(0.5,0.5,0.5);
    scene.add(torusMesh);
  }
  );
  camera.position.x = 20;
  camera.position.y = 20;
  camera.position.z = 80;
  document.body.appendChild( renderer.domElement );
  function renderAll()
  {
    requestAnimationFrame( renderAll );
  }
}
```

1. Three.js data

Displaying data with three.js requires the presence of renderer, a scene and a camera. The scene is a keeper element for the various objects - 3D artists refer to them as models - which are to be displayed in the scene. Cameras are a digital representation of the human eye looking into the scene and onto its contents from specified position. Finally the renderer is responsible for generating the picture seen by the virtual camera camera using mathematical processes.

2. 3D models

Displaying 3D scenes requires models and lighting. Our code starts out by using the JSONLoader on a predefined and relatively complex model created with Blender. When the loading is complete, the object is scaled to make it appear bigger and is provided with a set of position coordinates. Finally it is added to the scene, where it will be displayed later. In the next steps, similar processing takes place for the light source and the torus objects, which are to act as surroundings for the monkey.

3. Materials

The generation of the individual torii is interesting for multiple reasons. First, the corresponding geometry class generates a new skeleton of the desired element. This doughnut-shaped object is but a wireframe, bare of any physical appearance. Making it displayable is accomplished by adding a material. You should imagine MeshLambertMaterial to be like a skin of silk paper glued onto the torus. Then, a custom randomisation function is invoked to generate random coordinates.

4. Rendering

In the final step, the renderer is set up so that the scene is actually shown in the DOM tree. Here, the rendering must be requested in a relatively complicated fashion and that's because most browsers don't actually permit code to start rendering whenever it feels like doing so. Instead, a frame must first be requested for the rendering process to happen at all. A rendering handler will then be invoked when the graphics pipeline is ready to accept further data.



Power up HTTP

FIX HTTP AND REAL-TIME DATA EXCHANGE WITH WEBSOCKETS

Establish a connection

Communicating via WebSockets requires the presence of a connection. Fortunately, establishing one can be accomplished by creating a new instance of the WebSocket object. Pass in the URL or the IP address of the socket, and you're set to go.

```
function testWebSocket()
{
  websocket = new WebSocket(<url>);
}
```

Set up event handlers

Socket connections are asynchronous and event-driven by definition. Thus, the next step to WebSocket goodness involves the assignment of event handlers to the various members of the newly created WebSocket instance.

```
function testWebSocket()
websocket.onopen = function(evt) { onOpen(evt)
};
websocket.onclose = function(evt) {
onClose(evt) };
websocket.onmessage = function(evt) {
onMessage(evt) };
websocket.onerror = function(evt) {
onError(evt) };
```

Send some data

Transmitting information to the server is really easy: you can simply invoke the send function with a string or JSON object of your choice. Beware though that some browsers don't actually permit the transmission of objects, and so in that case, you can simply serialise them all by hand.

```
websocket.send("Hello World");
```

Receive server commands

Reacting to data coming from the server is as simple as using the onmessage event handler. Our snippet below prints the data to the console - when we are transmitting objects, the next step would be a classic application for deserialisation.

```
websocket.onmessage = function (event) {
  console.log(event.data);
}
```

Roll your own

WebSockets are supported by most server side development tools. Both Node.JS and Qt can be made to provide a WebSocket server right out of the box. Furthermore, third-party client libraries are available for all the major operating systems.

RESOURCES

THREE ESSENTIALS TO KEEP YOU CODING LIKE AN EXPERT

Website HTML5 Index html5index.org

The HTML5 Index is to JavaScript what the MSDN is to Windows Phone: a never-ending selection of interesting tid-bits, resources and other interesting stuff. The Index has helpful links on each term which corresponds with other technical terms. It even separates the terms by library for easy reading. It furthermore contains a detailed list of parameters and official API specifications which simplify coding.

YouTube channel Google Developers youtube.com/user/ GoogleDevelopers

Even though the big G's channel can become quite Android-centric at times, it nevertheless provides an attractive selection of amusing videos teaching new technologies and tricks of the trade. There are informative tutorial series like Developer How To or there are videos more aimed at the latest updates available like the 2015 I/O keynote speech. Be prepared for a deluge of amusing content - but watch that data cap if you're watching on mobile!

Twitter MozAppsDev twitter.com/mozappsdev

Mozilla's developer advocates are a never-ending source of all kinds of interesting stuff on the open web. Be it a tutorial, an interview, some musings on a RFC draft, a feature on a new standard or library: joining their 30,000 followers is a decision which you will surely not regret. They'll even advertise for job vacancies on there if you think you'll fit the bill.

Graceful degradation

When working on end-user-facing applications, an increase in compatibility tends to lead to a direct increase in earnings. Mobile developers have approached the problem of new APIs with gradual degradation. It is a straightforward technique based on the assumption that some features are not entirely necessary for application execution. Smart apps and websites offer their users a reduced featureset when run on legacy browsers.

Lock your mouse!

Pointer Lock solves the problem of being unable to control the cursor fully in web games. Using this snippet 'standardises' the vendor-specific property names with a common field: canvas.requestPointerLock = canvas.requestPointerLock || canvas.mozRequestPointerLock || canvas.webkitRequestPointerLock; This needs a call to requestPointerLock: canvas.onclick = function() { canvas.requestPointerLock(); }

Make it talk!

Really cool computers like Kit talk to their owners. The Speech API pulls off a similar trick, best accomplished like this: var sayWhat = new SpeechSynthesisUtterance("Let us say something!"); var voiceArr = window.speechSynthesis.getVoices(); sayWhat.voice = voiceArr[10]; sayWhat.pitch = 4; sayWhat.rate = 10; window.speechSynthesis.speak(sayWhat);

Listen up, Scotty

Some browsers also go the other way around using a proprietary server provided by Google. Their speech synthesis API usually takes a grammar, which is then analysed against the aural input. This mode cannot be used for free dictation - it is limited to formatted attributes from a known list. Due to the grammar in the background, it tends to be more reliable than free speech mode: entering village names, part IDs or airport codes should never be attempted in 'free scanning'.

GOOGLE ANALYTICS

Google Analytics is the most widely used web analytics service in the world and it's free. Find out the most important parts of the data to study and master key features to get more hits



“ Google Analytics is the window into your site’s data. Use it to understand who is visiting, what they are looking at, and what you can do to get them to stay longer ”

Know your audience

YOUR FIRST PORT OF CALL FOR GETTING INSIGHTS INTO YOUR AUDIENCE AND VISITOR BEHAVIOUR SHOULD BE THE ACQUISITION>ALL TRAFFIC CHANNELS REPORT

The Channels report enables you to see at a glance where your visitors are coming from according to Google's rules. Not all your traffic

will be captured by Google automatically, specific traffic such as Email will not automatically go under the Email channel.

Google uses their Direct channel as a catch-all mechanism for traffic that it just cannot categorise in any of its other channels. So it's important that you familiarise yourself with the UTM tags in order to make sure all campaigns you are running are accounted for.

To understand how Google uses these tags, it's important to get familiar with the concept of source - this could be Google, Bing, Facebook; and the concept of medium

- for example, paid search CPC, organic and so on. Google then uses rules based on these dimensions (not restricted to the ones mentioned) to define their channels.

For example the organic search channel is defined by all traffic that has a 'medium' exactly matching 'organic'.

The default channel grouping Google provides to you would cover most business' needs, but if you think you need a custom channel, Google enables you to create custom groupings based on these specific needs of yours. Just like the default ones, the custom channels are based on rules that you will have already defined in the interface utilising dimensions set out by Google Analytics.

Understanding key metrics

GET TO GRIPS WITH THE TERMINOLOGY USED IN THE GOOGLE ANALYTICS INTERFACE

Channels

Google's Default channel grouping splits traffic into eight groups and these are: Direct, Organic, Referral, Paid Search, Other Advertising, Email, Social and Display. You cannot change how these channels are defined but Google may evolve them in future. By using these groupings you will be able to clearly review visitor behaviour from a particular source and determine the channel's effectiveness without being muddled up by another channel.

UTM tags

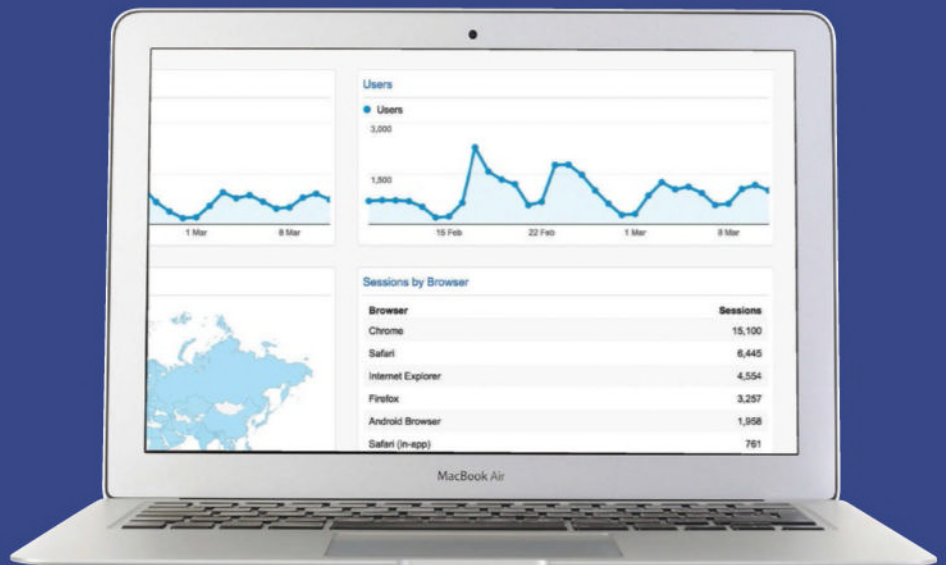
If you're planning some activity that will drive traffic to your site - email or social for example, it's key that you identify this traffic and make sure you can understand and measure the yield of your efforts. Google uses custom campaign parameters called UTM tags to enable you to identify each piece of activity or 'campaign'. For example, these identifiable activities can include the source, the keyword or term used, the content and ads or the specific name of a brand.

By tagging the links to your posts, for example in an email newsletter, you can ensure that this traffic is correctly identified as it comes through. Google has a handy URL builder tool that makes it very easy to identify the most effective URLs.

Real-time

Once you have tagged your campaign URLs you can test everything is working as it should be by looking at the Real Time analytics report by going to Real Time>Traffic Sources. By clicking through to your site using your newly tagged link, you should see this visit showing the correct medium and source in the report if you've done this right.

You will then be able to monitor spikes and trends as they happen on your website.



Sessions

The number of periods in which any user interacts with your website. If a user leaves the site and returns after 30 minutes or more, a new session is recorded.

Users

The number of unique users who have visited your site for at least one session during the selected date range.

Bounce rate

The proportion of total sessions which consist of only a single pageview being generated before the user leaves your website.

Pageviews

The total number of page impressions during user sessions within the selected date range.

Explaining relationship

A session consists of a grouping of one or more pageviews or other interactions which take place on your site. A session ends once a user has been inactive for 30 minutes, or leaves your site and returns via a different channel.

In-page analytics

Accessed from the Behavior section, In-Page Analytics offers

insight into how users navigate around your site. A live website overlay indicates the proportion of clicks made from any page to each linked page, enabling you to establish the most popular paths taken to explore your content.

Audience behaviour

The Audience>Behavior report reveals how loyal your visitors are and how often they return to the site. It's possible to compare the behaviour and performance of new versus returning visitors and also see the number of sessions each user generated during the selected date range.

Best new features

GOOGLE ANALYTICS IS REGULARLY UPDATED WITH NEW FEATURES AND REPORTS, HERE ARE SOME RECENT ADDITIONS THAT YOU NEED TO KNOW ABOUT

Spreadsheets add-on

This handy Google Analytics add-on for Google Spreadsheets enables easy access to your data via the Analytics API. It's possible to build your own reports that query multiple Analytics views and manipulate the returned data.

With a little setup you can create your own regularly updated custom dashboards merging Analytics data with any other measurements and making use of the Google Spreadsheets charts to visualise the results.

Cohort analysis

Currently in beta and rolling out across Google Analytics accounts, Cohort Analysis splits your audience into distinct groups. These are based on a particular behaviour or attribute and enables deeper level analysis. There is no such thing as an 'average user' and at present much analysis does not distinguish between differing behaviours, instead bundling all users into a single pot. Each grouping, or cohort, shares common characteristics and new trends may be unearthed.

Benchmarking

The Audience Benchmarking reports enable your performance to be compared against aggregate data from other sites of a similar size in the industry and country. You'll be able to see how your traffic sources and engagement metrics compare to other sites in the sector and discover areas where you outperform your competitors and those where you underperform. To access benchmarking reports you need to agree to share your data anonymously with Google.

Audience groups

Google's DoubleClick advertising technology collects anonymous information about web surfers and shares this in Google Analytics. This enables you to discover more about who is actually using your website. It's possible to view an estimated age and gender breakdown of site visitors, user affinities and the products and services users are seeking through In-Market Segments. Your audience can be segmented by these filters to enable you to find your most valuable user types.

Content Experiments

Content Experiments enables easy webpage split-testing which can help when working towards improving the performance of your website. When setting up a content experiment you must define a goal or conversion metric for which you are looking to enhance. For example, an eCommerce site could monitor revenue or the total number of transactions, whilst a lead generation site could look to increase the number of overall completed form fills.

Intelligence Events

Intelligence Events are a commonly overlooked feature. Analytics continuously checks for metrics that have seen statistically significant variations over recent days or weeks and highlights these anomalies. Intelligence Events help to surface any unexpected changes which could be indicative of a problem or a success on a particular page. It's also possible to set up automated email or text alerts which are triggered when a particular metric changes outside a defined threshold.

Plan your content

When it comes to creating content for your audience, making a decision can take up time. Luckily, it doesn't have to be this way.

Best performing existing content: pageviews

Use the 'All pages report' to check your most popular posts. If one topic outstrips the others in pageviews then see if you can address another aspect of the same topic.

Most engaging existing content: time on page

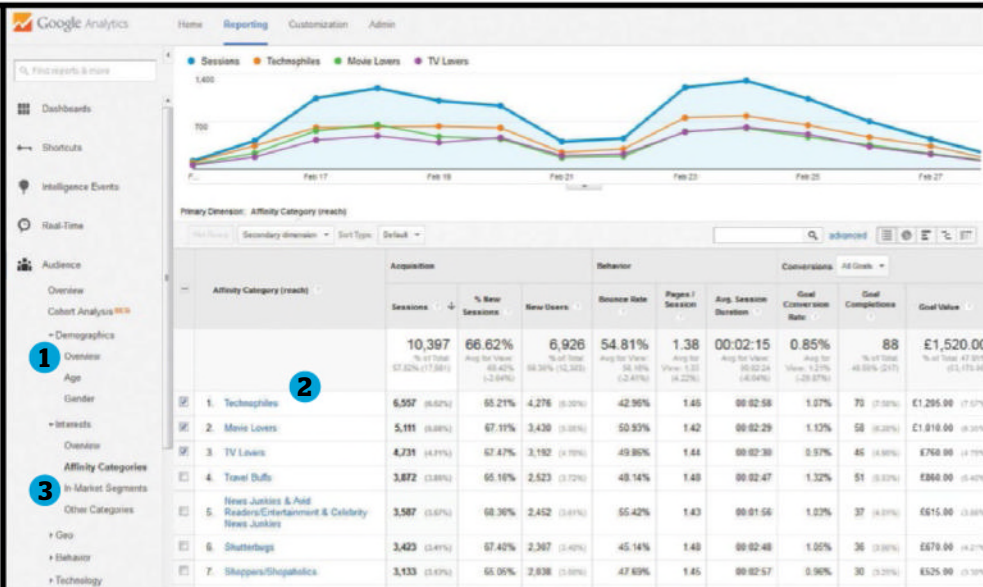
If you find an unusually high time spent on a post then write another with a similar angle, and interlink for the same effect. Use 'All pages report' again to find this out.

Internal site search

This one is straightforward. Found in the Behavior tab, the site search lets you know what your visitors have been looking for within your website.

Affinity Categories

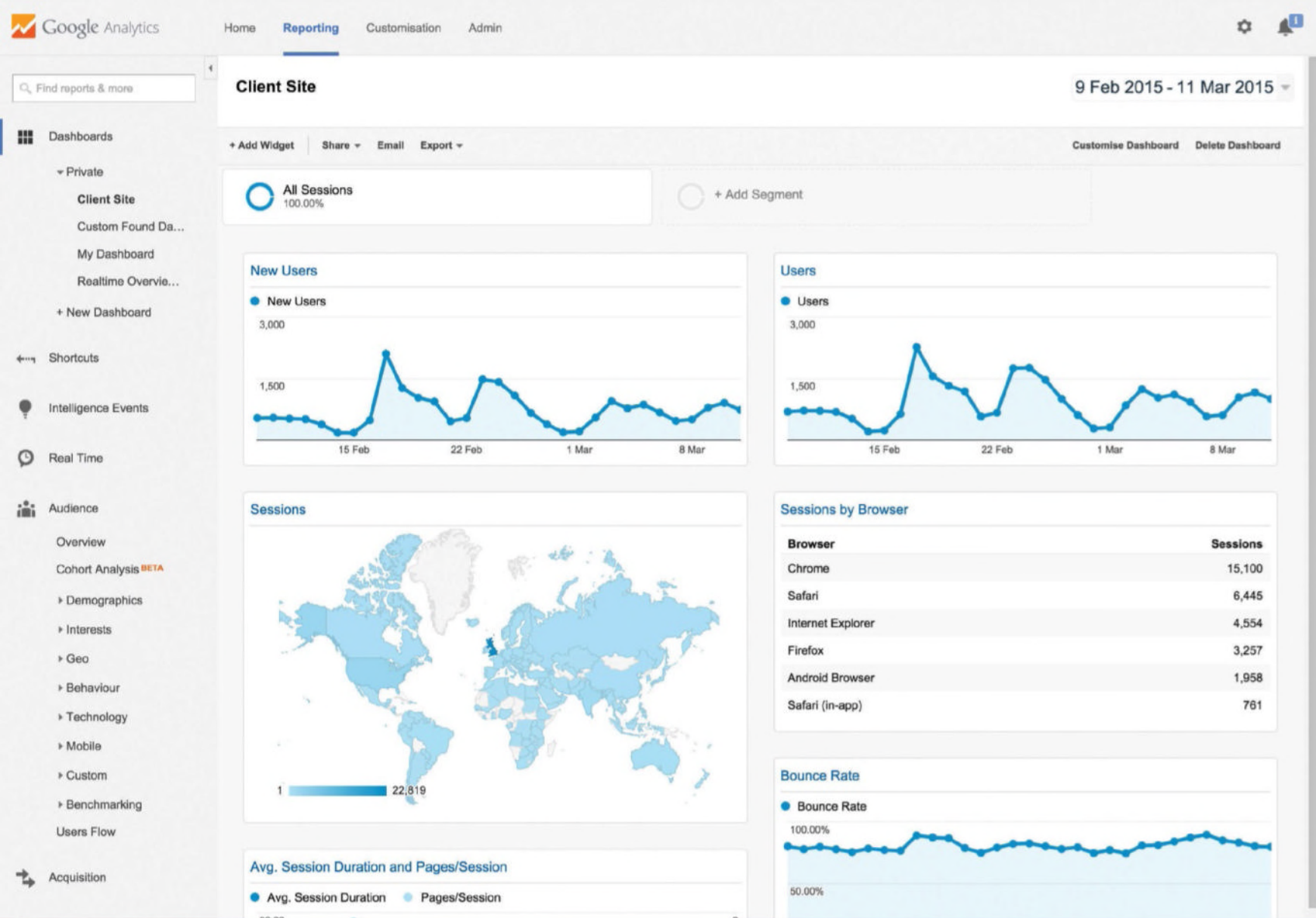
The Affinity Categories report can reveal your visitors' strongest interests. Including a post on a related interest and referencing effectively can provide new engagement.



1. Demographic reports can be viewed to reveal your visitors' data, like their gender split and age groupings.

2. User Affinity Categories can give insight into what else your users do online. Affinities reveal a more detailed interest breakdown.

3. The In-Market Segments report highlights areas of commercial intent in which your users have expressed an interest.



Customise the Google Analytics UI

Intelligence Events and custom alerts

GA generates automatic web alerts whenever there is a significant change in usage or traffic metrics. It can generate AdWords alerts in the GA interface if your accounts are linked. You can also set up custom alerts to be generated when traffic reaches or decreases to a certain threshold. Google enables you to receive these alerts via email or text message so that you can stay in the know even though you are not checking your GA reports.

There are four reports Google has available in terms of Intelligence Events: Overview, Daily Events, Weekly Events and Monthly events.

There are also some useful alerts you can set up like, if revenue drops by 10 per cent, if a landing page's bounce rate increases by 20 per cent, or if a PPC campaign performance increases revenue by 10 per cent, for example.

Custom dashboards

The usefulness of GA goes beyond the standard reports they provide - the interface is completely customisable and enables you to create bespoke dashboards and reports to match your needs. While most of the default dashboard provided out of the box may suffice, there are countless widgets you can create to really enhance your data analysis. Along with this, Google provides a great dashboard, report and segment repository called the Solution Gallery - a crowdsourced area where Google Analytics users publish their own dashboards, reports, anything that they have found useful for the benefit of fellow GA fans. This is a great shortcut and a brilliant resource too as the uploaded shortcuts are rated by stars by other users. You can download any of these premade dashboards to your own account and start using them straight away for your own site analysis.

Custom reports

Just like custom dashboards, GA enables you to create custom reports which are a very powerful tool in drilling down into your site data. Here you will get to choose and select only the dimensions and metrics that you want to display. This is so that you can easily and quickly access the data relevant to you without spending ages clicking around in Google Analytics each and every time you want that kind of specific data.

Google's help section is very useful here as it has a handy list of all the dimensions and metrics that GA uses, and also gives hints on the valid dimensions or metrics combinations as not all of them can be queried together.

As mentioned, the Solutions Gallery has a large repository of well-made, crowdsourced custom reports from other Google Analytics users that you can easily download, utilise and customise for your website's needs.

Automated emails

It's very easy to set up GA so that it can send you customised reports via email at a frequency that suits. This will then save you from having to go in and generate them each time and ensures that you are comparing like for like each week or month. These emails also act as a handy prompt to go in and have a further dig around in your Google Analytics data so that you're always on top of the latest trends on your website.

Intelligence Events reports can also be set up to alert you with an email when certain criteria are met. If set up correctly these can be utilised as an extremely useful warning system that will alert you to problems with your site or server that may need urgent attention, like if the bounce rate or page load time has increased significantly for example. Alternatively, the reports can be set up to make you aware of large traffic spikes that may warrant further investigation.

Internal site search

If your site utilises an internal search function to display a filtered selection of products or list of related articles, GA can track and report on this data.

The Site Search report, accessed under the Behavior section, provides details of the most commonly searched for keywords and the pages from which users make such queries.

A quick look through these terms will enable you to identify the content that users are actively seeking on your site. If this content exists, its prominence should be improved, if not you have a list that is all ready to go with suggestions for future content additions.

Search queries

Connecting your Google Webmaster Tools account to Google Analytics enables Webmaster Tools' off-site data to be combined with the on-site data in GA and unlocks the Search Engine Optimisation series of reports. The

most useful of these reports, Acquisition>Search Engine Optimisation>Queries, lists the different Google search queries for which your site has appeared in results. The data returned shows an estimate of the number of times your site was shown for each query as well as the average ranking position and click-through rate.

With most keyword referral data now reported as not provided, this report can give you visibility on the search terms that are sending traffic.

Pinpoint the best

As well as identifying pages that are performing below par, it's important to understand and learn from the success of your best performing content.

If you can identify common themes in this content then similar content could be developed to bring in more traffic. The Behavior>Site Content>Landing Pages report reveals details of the sessions which result from a user landing via a certain channel.

Ecommerce and goal conversion data can be particularly insightful here as after landing on a particular page, users may go on to explore your site and convert at a different URL. It's possible that a page with a low average time on page may in fact be extremely valuable, for instance a category page which funnels users to products they then go on to purchase.

Determine the issues

IDENTIFY YOUR POOR-PERFORMING PAGES VIA PRESENTABLE DATA AND OPTIMISE

Bounce rate

Pages which have a high bounce rate are usually an indication that the content is not what the user expected to find. On eCommerce websites, product pages frequently have high bounce rates as users may land on them from a search engine when looking for a particular product. It is possible to decrease bounce rate on these kinds of pages by ensuring that all possible related products are interlinked.

A page that records a bounce rate significantly greater than other pages of the same type should be investigated further.

It's also possible to view bounce rate by channel, which may reveal users from social platforms are interested primarily in a single piece of content and less in exploring the rest of your site.

Time on page

As well as recording average session duration, GA also reports on the average duration of pageviews of each individual URL on your site. Pages can easily be sorted by the average time on page metric, and this can be accessed through Behavior>Site Content>All Pages to reveal the pages where users are leaving quickly. This data suggests that users do not find these pages engaging enough to stay on.

The Content Drilldown report aggregates page-level metrics by a site's URL directory structure, which can quickly highlight particular subfolders such as product categories that are under performing.

Low converting pages

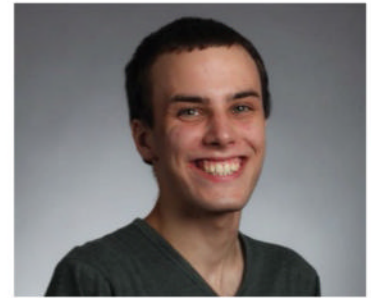
If you've set up eCommerce or goal-tracking then Google Analytics will be able to report on a true monetary value of a page. Selecting the eCommerce option at the top of Behavior>Site Content>Landing Page will reveal a report on the number of sessions, transactions and revenue that has been generated through visitors who have landed on a particular page that you want to analyse. It's also possible for you to view the average order value, eCommerce conversion rate and per session monetary value for that particular landing page, which can flag up any poor-performing pages for your attention. This will then enable easier fixing later on.

Selecting a Goal Set option at the top of the same report enables conversion rates for different predefined goals to easily be compared.

Pages with low click-through rates

The Acquisition>Search Engine Optimization>Queries report can highlight pages which rank highly in search results but aren't delivering much traffic. It will identify a troublesome query, for which your site occupies a high average position but has a low click-through rate and this will suggest that your listing is not as enticing as other competing sites at encouraging searchers to go ahead and click through to your website.

You can effortlessly improve upon this by updating webpage title tags and meta descriptions to ensure they contain the related search query and a strong call to action. This is a quick-and-easy optimisation that should increase click-through rate and result in additional traffic.



“If you're not a regular Google Analytics user, spending just half an hour or so looking through the reports could reveal some easy opportunities to boost traffic”

Philip Gamble
Technical SEO manager,
Found



“The Users Flow report is an overview of your best and worst performing pages. It's useful for gathering ideas on user journeys needing improvement”

Dora Moldovan
Head of technology, Found

PPC

If you are running a PPC campaign, it's important that you understand whether the traffic that is being generated is money well spent. By setting up goals in Analytics and assigning a monetary value to those goals, you can quickly establish the ROI on your PPC spend even if you aren't selling something, for example, a

subscription to your mailing list or a form submission can be a goal and assigned a value.

Remember that your PPC traffic should always have clear objectives and be directed at the most relevant landing page on your website. This will also need to be via the keywords that you are already using in order to give it the best chance at converting.



Social media

Social media traffic is altogether different from PPC. These visitors have generally arrived with far less intent to convert and have followed an interesting article or post that has led them to your site as oppose to an advert or offer. Understanding their behaviour however, is no less important. Do they

go on to read additional content on your site? Does a particular type of post generate more interactions than the others? Do they go on to find other content from your site and share that with their network of friends and followers? This information is invaluable for growing your audience and maximising the effectiveness of what you do for your site on social media and for building your brand.

Tracking your ROI - conversions

THE BEST WAY TO DETERMINE IF YOUR SITE IS SUCCESSFUL IS BY MEASURING YOUR GOALS

Once you set up the Google Analytics tracking code for your site, it's important to set up some conversion points for your site. Your business goals can vary - from tracking your online transactions to tracking whether someone has filled in your contact form, signed up to a newsletter or clicked to view a video.

Defining your conversions and understanding them in conjunction with your channel is key to your campaign success. For example, you may find that your paid search campaigns might be better at getting users to convert whereas your social media campaigns might be better at creating awareness of your brand and is a better traffic driver. It's important you define conversion points wisely and check your channels so you can understand your website or business' particularities and know where to channel your marketing efforts.

Types of conversions in GA

GA lets you set up different conversions for different needs. For example, Goals can be used for tracking key actions that occur on the site that usually reach a confirmation page, eg submitting a contact form, whereas Events is used to keep track of interactions that are actions typically tied in with page elements, like

clicking a button. GA provides a powerful API to enable merchants to track their website sales.

Goals

Google Analytics goals represent completed activities and conversions that measure your business objective success. Having properly defined goals enables GA to provide you with great insights on the effectiveness of your website page design or marketing campaigns. You can define up to 20 goals per reporting view.

GA lets you define your goals in several ways: Destination - where the goal is defined by reaching a specific page; Duration, a session that lasts a set amount of time or longer; Pages/Screens per session, where your goal might be to make sure the users see an x number of pages per session; and Event, where you can set up a goal based on an event, and this is useful if you want that event to be part of an attribution model like video play.

You can also define funnels for your goals and these are specific paths the users take to conversion. This is useful if you want to view where users abandon the funnel and where optimisation efforts should be spent.

Events

Events are user interactions that can be tracked independently from a page visit or load like video plays,

AJAX content, button clicks and so on. The difference between goals and events is that while goals are counted once per session, events are counted once per interaction. So if someone plays a video five times in a visit, that would register as five events. If they submit a form twice, that would count as a single goal. It is important that this difference is understood and the goals and conversions are implemented accordingly.

Another difference between goals and events is that the events need to be implemented programmatically via JavaScript code on the site, while goals can simply be defined in the GA interface. Using Google Tag manager makes goal implementation easy, so you should make sure that this is a consideration.

Ecommerce tracking

Based on information like products bought, transactions, and the time it takes for a user to purchase, you can get insights on your best-selling products or brands or categories, which channels are your best performers, your paid search ROI, and how long it takes customers to make a decision to purchase.

You can set up eCommerce tracking programmatically, but many kinds of shopping cart software will include GA eCommerce out of the box or provide easy-to-install extensions.

Glossary

Conversion

Any completed action that helps your business measure success. Conversions can be macro conversions or micro conversions depending on their importance. A completed purchase would be a macro conversion whereas something like a Facebook like would be a micro conversion.

Event

This is a type of user interaction with the page content. Events are utilised to track conversions when the interaction is independent on a page load. Examples of Events can include interactions like AJAX loaded content, link clicks, social buttons interactions, flash elements and so on.

Goal

A type of user interaction with the website. This is usually measured by the user reaching a specific URL or closely tied to a page load. Examples of this would be a user reaching some form of a 'Thank you' page, a purchase completed or a specified amount of time spent on a specific page.

Segment

A subset of sessions or users that share common attributes. Segments enable you to isolate and analyse groups of sessions or users for better analysis. You can apply up to four of these at a time and you can use predefined segments or import them from the Analytics Solution Gallery.

Channel grouping

A roll-up of traffic sources in the Acquisition reports that groups several marketing activities together. Channel groupings let you compare aggregated metrics by channel name, individual traffic source, medium or campaign name.

Attribution model

An attribution model determines which channels your sales and conversions get assigned to. This is based on contribution and there are many user models, for example a Last Interaction model will assign the credit to the last click before its conversion.