Media queries CSS3

modifié le : 16/01/2015

# Quelques définitions

**Application native**

Correspond aux logiciels à télécharger que l'on peut trouver sur les boutiques "stores" (AppStore pour Apple, GooglePlay pour Android,…) développés dans un langage spécifique à chaque plateforme : Objective-C pour Apple, Java pour Android, …

**Application web (Web App)**

Une application web est le nom que l'on donne à un site web "classique" que l'on a adapté pour les mobiles.

La structure existante reste en place (HTML, scripts, bases de données, médias) et est complétée par une couche graphique dynamique (en CSS) adaptée aux écrans de taille réduite dans un principe général de design adaptatif (Responsive web design et avec des outils CSS3 tels que les Media Queries. Des retouches ou sur-couches JavaScript sont parfois également à l'ordre du jour.

**Responsive web design**

Ensemble de techniques et principes permettant d'adapter un design web à différentes tailles d'écran : de l'écran de bureau géant au smartphone, en passant par les tablettes, les netbooks et autres formats de TV. Ceci afin d'offrir aux visiteurs la meilleure expérience possible quel que soit le support de consultation et quitte à réorganiser l'affichage et la disposition des éléments (par exemple passer d'un site à 4 colonnes sur grand écran à un site mono-colonne sur mobile).

Ce terme a été introduit par Ethan Marcotte dans son livre éponyme et se fonde sur un trio de techniques : CSS Media Queries, gabarits fluides et contenus fluides.

**Site web dédié**

Il s'agit d'une nouvelle version du site web complètement adaptée aux terminaux nomades en terme de technologies (HTML, vidéo, audio, images) et de performances (optimisations, préchargements, mises en cache). Habituellement, une détection du périphérique utilisateur (via User Agent, JavaScript, autre) permet de le rediriger vers l'URL de la version mobile dédiée du site (m.monsite.com, monsite.mobi).

**Navigateurs mobiles**

Le marché des navigateurs pour mobiles se distingue de celui des ordinateurs de bureau par sa diversité et sa vivacité concurrentielle et technologique.

Parmi les plus employés dans le monde : Opera Mobile, Opera Mini, Safari Mobile (sur iPhone, iPad), navigateurs Android, Internet Explorer mobile, Blackberry, Firefox mobile, Chrome mobile, UCweb, etc.

**Densité de pixels**

Correspond à la résolution d'un écran numérique exprimée en DPI (Dots Per Inch : points par pouce) ou PPI (pixels par pouce). Une densité élevée correspond à un écran dit "HD" ou "Haute Définition". Quelques exemples de densités : 326 PPI pour iPhone 4 et 4S, 264 PPI pour iPad 3, 220 PPI pour MacBook Pro 3.

**Pixel ratio**

Correspond au ratio entre les pixels réels (screen width) et la valeur de device-width du navigateur. Par exemple, un iPhone 4 a un ratio de 2 (640/320) tandis qu'un iPhone 3 a un ratio de 1 (320/320). Il est possible de détecter le pixel-ratio via Media Queries ou en JavaScript via window.devicePixelRatio.

**Retina**

"Retina display" est une marque déposée par Apple pour désigner des écrans dont la densité de pixels est telle que l'oeil humain n'est plus censé pouvoir distinguer de défauts dans les détails. Il ne s'agit ni plus ni moins d'écrans de haute définition (densité de pixels élevés). Cette technologie équipe des terminaux iPod, iPhone, iPad et MacBook Pro. Du côté de la concurrence, par exemple Samsung, on retrouve le terme (Super-)AMOLED.

**Orientation**

L'orientation correspond, pour un écran numérique, à son sens d'affichage : si le périphérique est tenu dans le sens de la largeur, on parle d'orientation paysage (landscape en anglais); s'il est tenu dans le sens de la hauteur, on parle d'orientation portrait. Les récents navigateurs de bureau reconnaissent également l'orientation : si la fenêtre est plus large que haute, l'orientation sera définie en paysage par exemple. Il est possible de détecter l'orientation via CSS Media Queries.

**Media queries**

Module des spécifications CSS3 permettant la sélection fine de périphériques selon divers critères (largeur, hauteur, résolution, orientation,…), puis d'appliquer spécifiquement un ensemble de styles CSS dédiés à cette sélection.

Cet outil est devenu une étape incontournable de l'adaptation de designs aux écrans de différentes tailles. Pour tous les détails,

**Point de rupture**

Les points de rupture sont les différents palliers, qui une fois franchis, "cassent" votre design, les largeurs qui nécessiteront des adaptations voire des réorganisations en CSS. Testez votre site web sous différentes résolutions (par exemple avec les excellents outils Screenqueri.es ou Responsive.is), vous identifiez rapidement quelques palliers problématiques : de sont vos points de rupture.

**Mobile first**

Philosophie de conception à contre-courant de nos principes habituels consistant à commencer prioritairement par la version épurée (mobile) puis à enrichir progressivement pour les écrans de plus en plus larges.

**User agent**

L'agent utilisateur est une chaîne de texte envoyée par le navigateur au serveur pour s'identifier. Elle contient des informations comme par exemple : le nom de l'application, la version, le système d'exploitation, la langue, etc. Il est possible, via un langage serveur, de récupérer et exploiter ces informations, par exemple en redirigeant le visiteur sur une version mobile dédiée en reconnaissant un agent utilisateur mobile. La technique n'est malheureusement pas toujours fiable car il existe des milliers de chaînes d'identification différentes et peu de sites les maintiennent à jour, ou les gèrent avec exhaustivité.

# Notions importantes

**résolution d'écran (définition)**Nombre de pixels réels de l'écran

**device**   
Nombre de pixels CSS virtuel de l'écran pour l'affichage (déclaré par le périphérique)

**viewport**   
Nombre de pixels déclarés par le navigateur. On se base sur lui pour l’affichage mais il ne correspond pas à la réalité

En général les 3 sont différents !

**screen width (ou screen height)**

Correspond à la largeur (ou hauteur) d'un écran numérique. Que cela soit sur un écran de bureau, une tablette ou un téléphone mobile, cela se traduit par la (mal nommée) résolution : par exemple 1024x768, 320x640, etc.

**device-width (ou device-height)**

Correspond aussi à la largeur (ou hauteur) d'un écran numérique. Ou plutôt celle qui est déclarée par le périphérique. Un certain nombre de terminaux mentent sur la valeur de device-width et device-height. En effet, ceux-ci ne correspondent pas à la largeur réelle de l'écran (screen-width) mais à une valeur exprimée en "Device Independant Pixels (dip)" ou "Pixels Indépendants du Terminal". Ainsi, un iPhone 4 dont la largeur physique réelle est de 640px retournera une largeur device-width de 320px, ceci pour des raisons de compatibilité ascendante avec son aîné iPhone 3.

**viewport**

Désigne la zone de la fenêtre du navigateur. Les navigateurs mobiles trichent en reconnaissant par défaut une surface de viewport plus élevée que la surface physique réelle en pixels. Par exemple la largeur de viewport par défaut sur Safari mobile est de 980px, elle est de 850px sur Opera, 800px sur un navigateur Android, 1024px sur IE mobile. Pour s'affranchir des tailles de sites minuscules (puisque le mobile affiche une largeur de 980px dans une surface de 320px), il est possible de modifier cette valeur par défaut et d'imposer ses valeurs à l'aide de l'élément HTML - propriétaire - <meta name="viewport"…>, ou selon les spécifications CSS, à l'aide d'une règle-at @viewport.

Pour résoudre ce problème d’affichage Apple a introduit cette balise HTML qui est maintenant adopté par la grande majorité des navigateurs récents

**<meta name="viewport" content="width=device-width,initial-scale=1.0" />**

Le W3C a aussi une norme CSS3 pour palier à ce problème mais à ce jour elle n’est pas supporté par tous les navigateurs mobiles

@viewport {

width: device-width; /\* largeur du viewport = largeur virtuelle pour affichage CSS \*/

zoom: 1; /\* zoom initial à  1.0 pour éviter que le navigateur dé-zoom la page au chargement \*/

}

@-webkit-viewport{

width: device-width;

}

@-moz-viewport{

width: device-width;

}

@-ms-viewport{

width: device-width;

}

@-o-viewport{

width: device-width;

}

@viewport{

width: device-width;

}

\*/

/\*

Par défaut la taille du viewport d'un terminal mobile ne correspond ni à la taille de son écran réelle ni celle en "pixels CSS".

Elle est généralement bien supérieure à la surface physique, afin de pouvoir y caler n'importe quelle page web en lui affectant un niveau de (dé)zoom.

La valeur initiale du viewport ne dépend pas du terminal, comme on pourrait le supposer, mais...

du navigateur mobile (et peut parfois même être modifiable par l'utilisateur dans ses réglages).

Voici quelques valeurs par défaut :

Android 1, 2 et 3 : 800px

Android 4 : 980px

Opera mini : 850px

Opera mobile : 980px

Safari mobile : 980px

Internet Explorer mobile : 1024px

---------------- Niveau de zoom initial --------------------------------------

Compte tenu de ces différentes surfaces, les pages web s'affichent par défaut de manière à ce que toute la surface entre dans celle de l'écran.

Ce niveau de zoom initial correspond à une simple division mathématique de device-width / viewport.

Au final, la surface réelle en pixel n'est pas prise en compte dans le calcul du niveau de zoom d'affichage.

Par exemple, Safari mobile sur iPhone 4 va afficher par défaut les pages web dans une fenêtre de 980px de large au sein des 320px de largeur qu'il croit avoir…

bien qu'il en ait physiquement 640px. Le niveau de zoom initial sera de 320 / 980, soit environ 0.326x.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*device\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**Attention, les terminaux Apple ont une particularité que n'ont pas les autres mobiles :**

**la valeur de device-width est invariable en portrait et en paysage.**

Le device-width d'un iPad vaudra toujours 768px quelle que soit l'orientation.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*media query\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Cette règle est actuellement déjà implémentée sur Opera mobile, IE10 mobile et Firefox mobile, et se présente sous la forme suivante :

**Opérateurs possibles : and only not**

Pour obtenir l'équivalent du "ou", il suffit d'énumérer différentes media queries à la suite,

séparées par des virgules : si l'une d'entre elles est valable, alors l'ensemble de la règle sera appliquée

**Fonctionnalités**

La plupart des critères (ou fonctionnalités) peuvent être préfixés par min- et max-

Lorsqu’elles acceptent des valeurs numériques pour définir des valeurs minimales ou maximales à respecter.

color : support de la couleur (bits/pixel)

color-index : périphérique utilisant une table de couleurs indexées

device-aspect-ratio : ratio du périphérique de sortie (par exemple 16/9)

aspect-ratio : ratio de la zone d'affichage

device-height : dimension en hauteur du périphérique

device-width : dimension en largeur du périphérique

grid : périphérique bitmap ou grille (ex : lcd)

height : dimension en hauteur de la zone d'affichage

monochrome : périphérique monochrome ou niveaux de gris (bits/pixel)

orientation : orientation du périphérique (portait ou landscape)

resolution : résolution du périphérique (en dpi, dppx, ou dpcm)

scan : type de balayage des téléviseurs (progressive ou interlace)

width : dimension en largeur de la zone d'affichage

Les dimensions pourront être évaluées avec des unités (px, em).

Les ratio avec des fractions (entier/entier). Une résolution sera définie en dpi (points par pouce) ou en dpcm (points par centimètres).

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

@media screen and (min-width:960px) {

body {background-color:blue;}

#content {

background-color:yellow;

}

#content a {

color:#000

}

}

@media screen and (max-width:960px) {

body {background-color:red;}

#content {

background-color:blue;

}

#content a {

color:#FFF

}

}

@media screen and (max-width:768px) {

body {background-color:blue;}

}

@media screen and (max-width:550px) {

body {background-color:green;}

}

@media screen and (max-width:320px) {

body {background-color:orange;}

}

/\*

Autres combinaisons

\*/

/\* largeur entre 200 et 640px \*/

@media screen and (min-width: 200px) and (max-width: 640px) {

.bloc {

display:block;

clear:both;

}

}

/\* orientation:portrait \*/

@media screen and (orientation:portrait) {

.portrait {

background: #f11a57;

color:#fff;

}

.precisions span.regle5 {

display: block;

}

}

/\* orientation:paysage \*/

@media screen and (orientation:landscape) {

.landscape {

background: #f11a57;

color:#fff;

}

.precisions span.regle6 {

display: block;

}

}

/\* Pour l'impression \*/

@media print {

}

/\* pour la TV \*/

@media tv, (device-aspect-ratio: 16/9), (device-aspect-ratio: 16/10) {

}

/\*

@media (min-width: 700px) and (orientation: landscape) {…}

@media tv and (min-width: 700px) and (orientation: landscape) {…}

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Navigateurs\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*

Les navigateurs sont inégaux face aux media queries, mais cela tend à s'améliorer.

Mozilla Firefox : 3.5+

Internet Explorer : 9+

Google Chrome : 5+

Opera : 10.5+, Opera Mobile : 10+, Opera Mini : 5+

Apple Safari : 4+ et iOS (mobile) 3.2+

Android : 2.1+

WebKit en général

Globalement, si un navigateur ne les supporte pas du tout,

il continuera à appliquer la règle concernant le type de média qui figure en début de chaîne.

\*/

</style>

</head>

<body>

<div id="content">

<ul>

<li><a href="http://www.alsacreations.com/article/lire/1490-comprendre-le-viewport-dans-le-web-mobile.html">Comprendre-le-viewport-dans-le-web-mobile</a></li>

<li><a href="http://media-queries.aliasdmc.fr/meta\_viewport\_et\_viewport\_css\_mobile.php">meta viewport et viewport css mobile</a></li>

<li><a href="http://dev.w3.org/csswg/css-device-adapt/">CSS3 @viewport : W3C</a></li>

<li><a href="http://www.alsacreations.com/article/lire/930-css3-media-queries.html">Media-queries</a></li>

<li><a href="http://www.w3schools.com/cssref/css3\_pr\_mediaquery.asp">W3C mediaquery</a></li>

<li><a href="http://media-queries.aliasdmc.fr/media-queries-live.php">Media-queries-live.php</a></li>

<li><a href="https://developer.mozilla.org/fr/docs/Web/CSS/Media\_queries">Media-queries +</a></li>

<li><a href="http://mydevice.io/devices/">Liste des max-device-width et max-device-height (dimensions réellement prises en compte pour l'affichage) </a></li>

</ul>

<img src="proprietes\_@viewport.jpg" width="856" height="458" border=0 alt="">

</div>

</body>

</html>