

Responsive (web) design

*Créez les interfaces web une fois,
utilisez-les partout !*

CARRIÈRE

La fin du webmaster

Les métiers
et les salaires
du web

AGILITÉ

Scrum

au service du
développeur

- Découvrez le fonctionnement de la JVM
- Des interfaces dynamiques sous Android

- Redécouvrez Emacs
- Combiner Amazon EC2 et Amazon S3
- Les débordements de tampon

Printed in EU - Imprimé en UE - BELGIQUE 6,45 €
SUISSE 12 FS - LUXEMBOURG 6,45 € DOM Surf 6,90 €
Canada 8,95 \$ CAN - TOM 940 XPF - MAROC 50 DH

M 04319 - 162 - F: 5,95 €



Responsive Web Design : définitions et approches

Qu'est-ce qui se cache derrière le « Responsive Design » ? C'est en partant des fondations de cette nouvelle approche que l'on peut comprendre pourquoi et comment cela change complètement la manière de designer pour le Web. L'engouement autour de cette méthode est lié au changement du marché des périphériques pour consulter le Web, ce qui en fait un sujet incontournable pour les designers. Pour les intégrateurs, les solutions existent et sont simples à appréhender.

Elles remontent à un article de John Allsop écrit en 2000 dans lequel il présente sa vision du Web de demain <http://alistapart.com/article/dao>. Mais il faut attendre dix ans pour qu'une définition claire, code à l'appui, soit donnée. C'est ce qu'a fait Ethan Marcotte en mai 2010 dans son article « Responsive Web Design » (<http://alistapart.com/article/responsive-web-design>), dans lequel, comme il le déclarera lui-même plus tard (<http://youtu.be/EtOEutOOeeE>), il concrétise les notions abordées par John Allsop en 2000.

Flexibilité

Pour John Allsop, la flexibilité du format Web doit être vue comme un avantage et non plus comme une contrainte. Pour se faire comprendre, il utilise un rapprochement très intéressant entre la radio parente de la télévision et l'imprimerie parente du Web. Les premières émissions de télévision étaient simplement des émissions radio que l'on filmait. Partant de ce constat, la perspective concernant le média Web change : on pourrait croire que celui-ci est aujourd'hui mature, mais lorsque l'on observe le nombre de conventions qu'il emprunte encore à l'imprimerie sans recul, on revoit sa position sur le sujet. En effet, pour produire des interfaces facilement consultables par les utilisateurs, les designers se sont appuyés sur leurs compétences sur format papier. Or, celui-ci est par nature très strict : la taille des pages est définie et comme il est impossible d'interagir avec son contenu, celui-ci reste figé dans la page. Les designers ont donc pour habitude de travailler dans un cadre fixe : il s'agit de contrôler, pour chaque page, tout ce que perçoit et consulte l'utilisateur. Le designer attend alors des intégrateurs que ceux-ci reprennent exactement le modèle qu'il leur a donné. Il en est de même pour les clients qui ne comprennent pas toujours pourquoi le

rendu final de leur application n'est pas en tous points identique à ce que leur avait présenté le designer. Or, un des fantastiques avantages du Web, c'est le dynamisme possible du contenu de ses pages. Avantage que l'approche conventionnelle de l'expérience utilisateur ne permet pas de bien exploiter.

Adaptabilité

Ethan Marcotte reprend l'idée que présente John Allsop et lui donne une application concrète. Certaines contraintes techniques de 2000 ne sont plus aussi présentes, ce qui ouvre de nouvelles possibilités. Les navigateurs implémentent les nouveaux standards de présentation (CSS) avec toujours plus de soin. La dernière version de la norme CSS introduit les « media queries » qui permettent de spécifier le style à appliquer en fonction de certains critères d'affichage de la page Web : taille de la fenêtre, densité de pixel, etc. Ethan Marcotte en fait un des éléments principaux du « Responsive Web Design » permettant ainsi, enfin, la mise en place de l'adaptabilité des pages Web. Pourtant, en JavaScript, il est possible de récupérer ces critères depuis longtemps puis de changer l'affichage dynamiquement. L'utilisation des « media queries » est bien plus simple et propre mais n'offre pas de nouvelles possibilités. Pourquoi, alors, le Responsive Design et la réalisation d'interface de manière flexible et adaptative est-elle d'actualité ?

> LE CHOC SMARTPHONE

Afin de pouvoir travailler dans un cadre fixe, comme pour le format papier, il fallait convenir d'une taille cible puis de s'y contraindre. Cette démarche s'accompagnait généralement d'une étude pour connaître la taille des écrans les plus utilisés. Les utilisateurs minori-





Printgram



taires qui naviguaient sur un écran plus petit ou plus grand étaient alors tous simplement ignorés.

Une version spécifique

Avec l'arrivée des smartphones, le besoin exprimé a été dans un premier temps de réaliser des versions dédiées des applications et sites Web. Il s'agissait de la suite logique du « Specific Design ». La version « iPhone », les smartphones « Android » n'étant pas encore très présents sur le marché, devait suivre un design spécifique reprenant les normes et conventions du périphérique, un véritable casse-tête pour la mutualisation du contenu. De plus, les architectures en place proposaient rarement des services Web pour permettre à plusieurs applications d'interagir avec le système. L'arrivée des smartphones a donc introduit les prémices de la mise en place d'une architecture orientée services (SOA).

Et de deux

Le marché évoluant, la part de marché des utilisateurs ayant un smartphone sous Android ne pouvait plus être ignorée. Il fallait trouver une solution pour ces utilisateurs car les designs Web en place suivaient les préconisations d'Apple concernant la réalisation d'applications natives. Au-delà du style, la compatibilité surtout n'était pas optimale. Il y a donc eu une phase d'adaptation pour faire que la version mobile supporte à la fois les environnements « iOS » et « Android ». La charge de travail pouvait être assez conséquente et la plus-value perçue très minime. La stratégie utilisée pour aborder les smartphones commençait à être remise en question.

Et de trois ... vous êtes sûrs ?

La stratégie du « Specific Design » vola définitivement en éclats avec l'arrivée des tablettes et de Windows Phone. Il n'était plus question de produire encore une énième version du site ou de l'application. Si on additionne les coûts de développement des nouvelles solutions aux coûts de maintenance des anciennes, cette stratégie devient encore plus absurde. Car même pour des très grands noms qui n'ont pas de soucis de retour sur investissement, la complexité de gestion devient bien trop importante. Une solution à toutes ces problématiques est donc bien de suivre la nature de ce media qu'est le Web et de proposer des designs flexibles et adaptables comme l'ont suggéré John Allsop et Ethan Marcotte. Oui mais comment ?

> L'OPTIQUE MOBILE FIRST

Avant la phase d'intégration du design, qui utilisera alors nos fameuses « media queries », il faut réaliser une ergonomie et un design pour notre site. L'optique « Mobile First » consiste à se focaliser sur le format mobile en premier. C'est seulement ensuite que l'on fait

évoluer l'ensemble pour enrichir l'expérience utilisateur lorsque la surface d'affichage devient plus importante. Physiquement, les surfaces des mobiles sont très réduites. Le cadre que définit le mobile est alors très contraignant, il n'est pas possible d'afficher beaucoup d'informations simultanément. Le seul moyen de réussir à proposer une expérience utilisateur simple et efficace est de revenir à l'essentiel du besoin des utilisateurs. L'optique « Mobile First » permet donc à l'ensemble des acteurs autour de l'application, designers, ergonomes, commanditaires, de rester centrés sur les besoins premiers des utilisateurs et de bien les distinguer du superflu.

Ergonomie

L'ergonomie, ou expérience utilisateur, consiste à définir la manière dont les utilisateurs vont interagir avec le contenu du site. La séparation claire du design et de l'ergonomie est un facteur clé du succès de l'interface utilisateur. Cela se fait en réalisant des maquettes sans aucun esthétisme où les composants d'interface sont le plus abstrait possible. Le mieux est de procéder de manière itérative en réalisant des tests utilisateurs. Le livre « Don't Make Me Think! » de Steve Krug encourage les tests utilisateurs et est une excellente approche de l'ergonomie. Néanmoins, la flexibilité de l'ergonomie dépend de l'agencement des différents éléments de l'application entre eux, ce qui peut parfois devenir très complexe à mesure que l'on s'adresse à des écrans plus grands. En « Mobile First » la démarche est simplifiée car, dans un premier temps, on va surtout s'interroger sur ce qui doit vraiment être présent sur chaque page de site ou écran d'application. Les briques de base que sont les logo, titre, menu, article, produit, commentaire, etc., sont alors réalisées de manière très réduite et sont donc plus facilement identifiables, leur agencement devenant alors beaucoup plus aisé.

Pour la version mobile, les briques de base de la page seront généralement les unes au-dessus des autres. L'agencement des briques va ensuite être une affaire de colonage pour s'adapter à des tailles de page plus grandes. Les rubriques d'un menu qui étaient les unes au-dessus des autres se retrouveront les unes à côté des autres pour un affichage large. Le nombre de colonnes où les articles sont présentés passera d'une à trois selon la taille d'affichage.

Design

Encore une fois, lors de la réalisation du design, il est important de toujours bien séparer sa présentation avec celle de l'ergonomie auprès du client pour que des éléments d'esthétisme ne perturbent pas l'étude ergonomique. La phase de design doit elle aussi cibler le mobile. Comme pour l'ergonomie, le cadre minimaliste du mobile va contraindre le design à être simple et dépourvu d'éléments superflus. La phase de réflexion pour rendre le design flexible est plus déli-



cate. La définition des tailles des éléments va être très importante et souvent en rupture avec ce qui était proposé jusque-là. Les sites en « Responsive Design » vont avoir naturellement tendance à proposer des titres, textes et images plus gros et ayant moins de contenu. Le design est alors beaucoup plus focalisé sur la hiérarchisation de l'information pour faciliter son analyse que sur l'agencement d'une quantité trop importante d'information.

Graceful degradation

Pour réaliser des interfaces en Responsive Design, il existe une autre approche : le « Graceful degradation ». A l'inverse du « Mobile First », il s'agit de partir de l'interface la plus riche d'information (desktop) puis de supprimer des éléments pour arriver à la version la plus essentielle (mobile). Cette approche peut être intéressante dans le cas d'une refonte d'un existant. Il s'agit alors de vérifier si les éléments identifiés comme non essentiels sur la version minimale ont réellement une plus-value sur la version complète. La démarche est donc tout de même plus complexe qu'avec celle du « Mobile First » car il faut pouvoir être critique sur un travail déjà réalisé.

> EN PRATIQUE

Fluidité

Avant de cibler différents types d'écrans à l'aide des media queries, il faut réaliser un agencement fluide des différents éléments. Pour cela, la taille de ces différents éléments doit être définie le plus possible avec des pourcentages. Les images, les rubriques de navigation, etc., seront alors toujours présentées de manière optimale selon la taille de l'affichage. C'est une pratique en rupture avec les designs statiques car le rendu des pages ne semble alors plus contrôlé. La crainte de voir le design défini pour la page devenir inutilisable dans certains cas extrêmes d'affichage, pourrait être légitime si le « Responsive Design » s'arrêtait là. C'est alors qu'interviennent les media queries.

Media queries

Les medias queries sont dans les recommandations du W3C pour la norme HTML5/CSS3 depuis 2012. Le principe consiste à pouvoir appliquer des styles suivant un contexte d'affichage. Celui-ci va être ciblé en évaluant un ensemble de propriétés. Le conditionnement peut se faire lors de la déclaration de la feuille de style :

```
<link rel="stylesheet" media="screen and ("ma condition")" rel="stylesheet" href="exemple.css" />
```

Ou encore directement dans les styles :

```
@media screen and "mes conditions" { "mes css à appliquer" }
```

Le « et » logique entre plusieurs conditions se définit avec le terme « and ». Le « ou » logique se fait avec la virgule « , ». Attention toutefois, il n'est pas possible de mixer les deux en spécifiant des parenthèses. En d'autres termes : le « et » est prioritaire sur le « ou ». Il faut donc réaliser les expressions toujours avec les formes suivantes :

```
@media screen and ("ma condition") and ("ma condition") { "mes css à appliquer" }
@media screen and ("ma condition"), screen and ("ma condition") { "mes css à appliquer" }
```

Dans les exemples, le terme « screen » en début de requête permet d'adresser uniquement les périphériques de type « écran ». Un autre type communément utilisé est « print » pour les styles à appliquer

lors de l'« impression ». Il existe d'autres termes, mais ils ne sont pas actuellement implémentés de manière cohérente.

Les conditions

Elles se structurent par un couple de propriété/valeur séparées par « : ». Les propriétés sont constituées par des termes « width, height, etc. » qui peuvent se préfixer par « min » et « max ». Les préfixes servent à combler l'impossibilité d'utiliser les signes inférieurs et supérieurs « < », « > » car ils rentrent en conflit avec les balises HTML. Il est important de noter qu'il faut entourer les conditions avec des parenthèses.

```
@media screen and (min-width: 400px) and (max-width: 700px) { "mes css à appliquer" }
```

Les styles seront appliqués ici uniquement si la taille de la page est comprise entre 400 et 700 pixels. La liste complète des propriétés utilisables est disponible sur le site du W3C à l'adresse suivante : <http://www.w3.org/TR/css3-mediaqueries/>. Parmi les 13 propriétés de conditionnement, la propriété « width » sort du lot car elle permet de réaliser des conditions concernant la largeur de la page Web. Il s'agit de la propriété clé du « Responsive Design ». Cette propriété requiert de la subtilité car c'est elle qui permet d'adresser de manière optimisée les tablettes et smartphones. Le terme utilisé pour définir les paliers est « point de rupture ». Ce terme met bien en évidence le fait qu'ils servent à intervenir lorsque le design se « brise », c'est-à-dire lorsque le rendu n'est plus agréable. Il existe des points de ruptures standards, mais il est très important de garder en tête qu'ils doivent être redéfinis suivant le design choisi :

- Moins de 600px : smartphone
- 600 – 1024px : tablette
- Plus de 1200px : ordinateur de bureau

> FRAMEWORKS ET OUTILS

L'utilisation des frameworks de colonage permet de simplifier l'agencement lors des passages en mode mobile, tablette et desktop.

<http://foundation.zurb.com/>, <http://lessframework.com/>, <http://adapt.960.gs/>

Émulateurs

Au lieu de redimensionner la fenêtre de votre navigateur préféré, les émulateurs permettent de voir plusieurs formats en même temps ou de visualiser le rendu sur des périphériques cibles. <http://screenqueri.es/>, <http://quirktools.com/screenfly/>, <http://www.responsinator.com>

Mockup

Pour réaliser la phase d'ergonomie en faisant glisser les composants (textes, champs de saisie, etc.) dans un style abstrait.

<http://www.balsamiq.com/>

Exemples

Beaucoup d'exemples de sites en « Responsive Design » avec une présentation qui permet de visualiser le rendu selon 4 tailles d'affichage. <http://mediaqueri.es/>

Compatibilité

Certains scripts pallient l'absence des media queries lorsque les navigateurs n'implémentent pas les CSS3.

<https://github.com/scottjehl/Respond>,

<https://code.google.com/p/css3-mediaqueries-js/>

Guillaume Gaulard

Spécialiste Front-End : Ergonomie, Design et Développement, chez Zenika.

