

Les containers queries : une nouvelle façon de concevoir un site responsive

Par Thibaut Courant

le 28 Mars 2024





Concevoir un site « responsive »

58.22% du trafic internet mondial se fait sur mobile



Avoir un site responsive = = nécessité



Responsivité d'un site web quasi exclusivement gérée côté Front



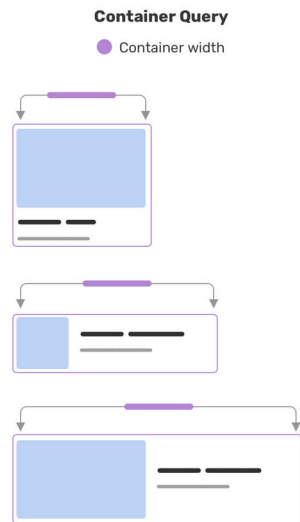
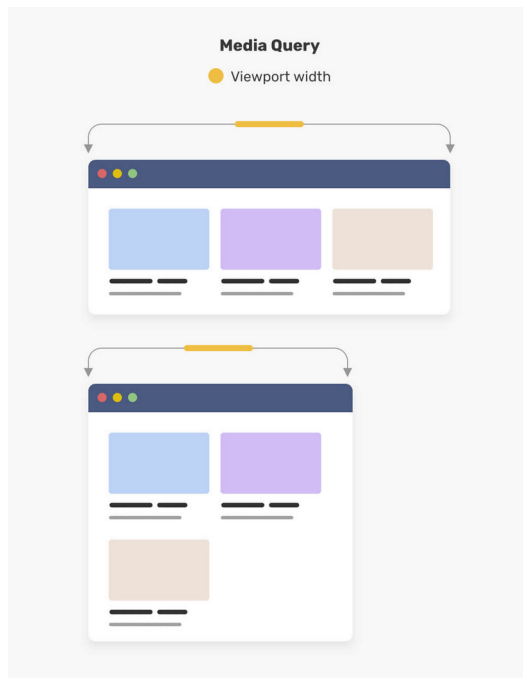
Container Queries vs Media Queries

Les Media Queries :

- Modifier l'apparence des éléments d'un site en fonction de **l'appareil ou du viewport**
- 30aine de propriétés de media utilisables (Taille / Couleur / préférences navigateur)
- @media width et @media resolution = > 99% des utilisations.



Container Queries et Media Queries



- Media queries :

La taille du viewport va conditionner l'affichage de **tous** les éléments du DOM

- Container queries :

Le container est **décontextualisé** :
Seule sa propre taille conditionne son affichage et celui de ses enfants



Container Queries vs Media Queries

Les Container Queries :

- **Atomic design : isoler chaque composant**
- Styliser les composants en fonction de leur environnement ou de leur parent proche

Chrome	Edge *	Safari	Firefox	Opera	IE
4-91					
92				10-78	
93-104	12-104			79-90	
105	105	3.1-15.6	2-109	91-93	
106-121	106-121	16.0-17.2	110-122	94-105	6-10
122	122	17.3	123	106	11
123-125		17.4-TP	124-126		



Supporté par 91% des navigateurs



Container Queries vs Media Queries

Les Container Queries :

- **Atomic design : isoler chaque composant**
- Styliser les composants en fonction de leur environnement ou de leur parent proche

Chrome	Edge *	Safari	Firefox	Opera	IE
4-91					
¹ 2 92				10-78	
¹ 93-104	12-104			¹ 79-90	
³ 105	³ 105	3.1-15.6	2-109	³ 91-93	
106-121	106-121	16.0-17.2	110-122	94-105	6-10
122	122	17.3	123	106	11
123-125		17.4-TP	124-126		



Supporté par 91% des navigateurs



Apporter de la flexibilité à la conception



Container Queries : présentation

Les Container Queries starter-kit :

- Déclaration

```
.card-container {  
  container-type: inline-size;  
  /* inline-size : la largeur du container */  
  /* size : va considérer la hauteur et la largeur du container */  
  /* normal : ne peut pas cibler les dimensions du container mais plutôt son style */  
  
  container-name: card;  
  
  @container card (inline-size < 500px) {  
  
  }  
}
```



Container Queries : présentation

Les Container Queries starter-kit :

- Déclaration

```
.card-container {  
  container-type: inline-size;  
  /* inline-size : la largeur du container */  
  /* size : va considérer la hauteur et la largeur du container */  
  /* normal : ne peut pas cibler les dimensions du container mais plutôt son style */  
  
  container-name: card;  
  
  @container card (inline-size < 500px) {  
  
  }  
}
```



La propriété à vérifier



Container Queries : présentation

Les Container Queries starter-kit :

- Déclaration

```
.card-container {  
  container-type: inline-size;  
  /* inline-size : la largeur du container */  
  /* size : va considérer la hauteur et la largeur du container */  
  /* normal : ne peut pas cibler les dimensions du container mais plutôt son style */  
  
  container-name: card;  
  
  @container card (inline-size < 500px) {  
  
  }  
}
```



La propriété à vérifier



Les breakpoints à définir

- Connaître l'état du container
Inline-size , *width*, *block-size*, *height*,
aspect-ratio, *orientation*



Container Queries : présentation

Les Container Queries starter-kit :

- Déclaration

```
.card-container {  
  container-type: inline-size;  
  /* inline-size : la largeur du container */  
  /* size : va considérer la hauteur et la largeur du container */  
  /* normal : ne peux pas cibler les dimensions du container mais plutôt son style */  
  
  container-name: card;  
  
  @container card (inline-size < 500px) {  
  
  }  
}
```



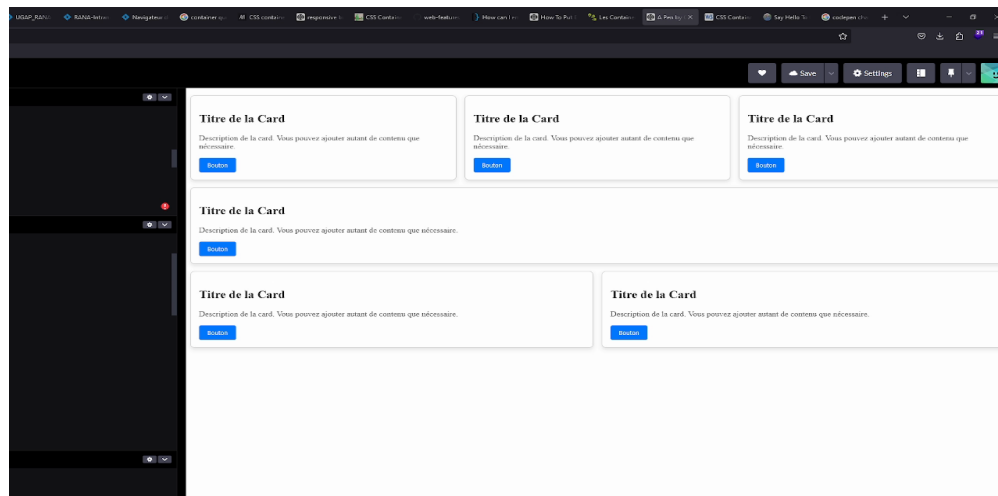


Container Queries : en pratique

```
card-container {
  width: 100%;
  container-type: inline-size;|
  container-name: card;

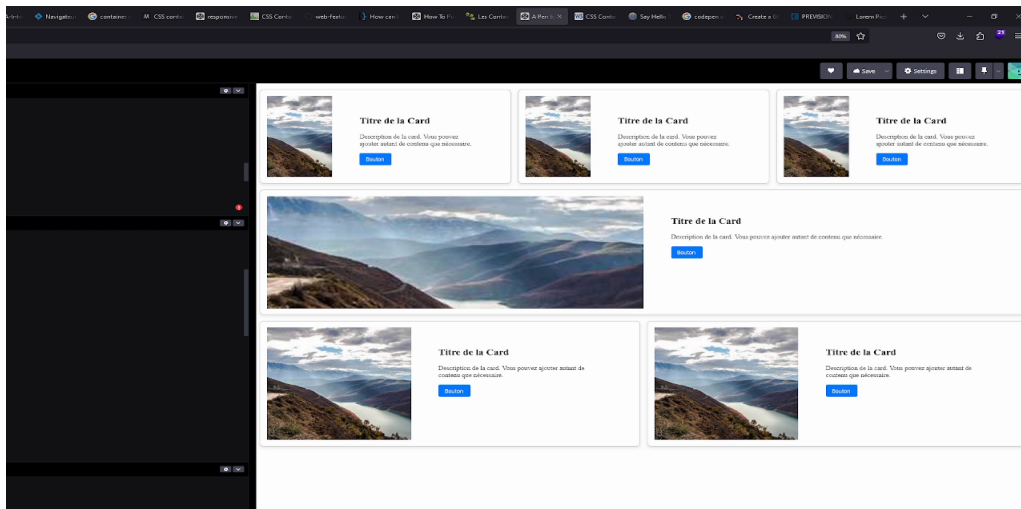
  @container card (inline-size < 500px) {
    .card {
      background: orange;
      h2,
      p {
        color: white;
      }
    }
  }

  @container card (inline-size < 250px) {
    p {
      display: none;
    }
  }
}
```





Container Queries : en pratique





Container Queries

➔ Moins de code que pour les Media Queries pour le même résultat



Container Queries

- ➔ Moins de code que pour les Media Queries pour le même résultat
- ➔ Indépendance des composants : facilite la réutilisation



Container Queries

- ➔ Moins de code que pour les Media Queries pour le même résultat
- ➔ Indépendance des composants : facilite la réutilisation
- ➔ Flexibilité des composants : gestion plus fine des propriétés css



Container Queries : unités

De nouvelles unités !

Cqw	1% de la largeur du container
Cqh	1% de la hauteur du container
Cqi	1% de la largeur inline
Cqb	1% de la hauteur de block
Cqmin	Plus petit cqi ou cqb
cqmax	Plus grand cqi ou cqb



Des unités plus précises
que les unités de viewport (vw et vh)



Container Queries : unités

De nouvelles unités !

Cqw	1% de la largeur du container
Cqh	1% de la hauteur du container
Cqi	1% de la largeur inline
Cqb	1% de la hauteur de block
Cqmin	Plus petit cqi ou cqb
cqmax	Plus grand cqi ou cqb



Des unités plus précises
que les unités de viewport (vw et vh)



Encore améliorer le responsive

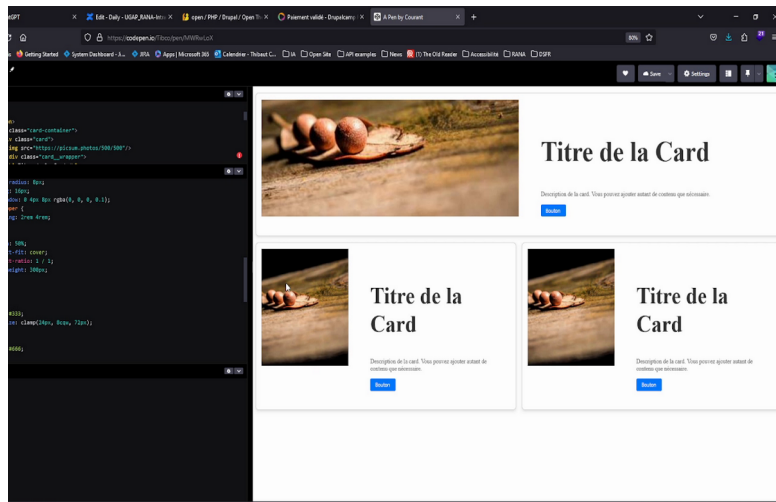


Container Queries : unités

Exemple : font-sizes responsives

```
h2 {  
  color: #333;  
  font-size: clamp(24px, 8cqw, 72px);  
}
```

Les container queries units peuvent être utilisées pour n'importe quel élément descendant du container déclaré





Containers Queries : Et pour Drupal ?

- Flexibilité dans l'utilisation de Layouts :

Layout Paragraph / Layout builder ✉ *Moins de code lorsqu'on réutiliser un paragraphe*



Containers Queries : Et pour Drupal ?

- Flexibilité dans l'utilisation de Layouts :

Layout Paragraph / Layout builder ✉ *Moins de code lorsqu'on réutiliser un paragraphe*

- Optimisation de l'atomic design

Single directory components ✉ *Associer son style directement à son module*



Containers Queries : Et pour Drupal ?

- Flexibilité dans l'utilisation de Layouts :

Layout Paragraph / Layout builder ✉ *Moins de code lorsqu'on réutiliser un paragraphe*

- Optimisation de l'atomic design

Single directory components ✉ *Associer son style directement à son module*

- Theming Drupal

Réutilisation plus facile du code en changeant de thème

Propriétés css natives : utilisation possible via scss, sass ou less compilé



Dois-je abandonner mes Media Queries ?

➔ Les container Queries sont complémentaires



Dois-je abandonner mes Media Queries ?

- ➔ Les container Queries sont complémentaires
- ➔ Les container Queries adaptées aux petits éléments (Cartes, tuiles, images, vidéos)



Dois-je abandonner mes Media Queries ?

- ➔ Les container Queries sont complémentaires
- ➔ Les container Queries adaptées aux petits éléments (Cartes, tuiles, images, videos)
- ➔ Media Queries plus adaptées à la grille et wrappers : `<body>` `<main>` `<section>` car leur taille est en général égale à celle du viewport.



Dois-je abandonner mes Media Queries ?

- ➔ Les container Queries sont complémentaires
- ➔ Les container Queries adaptées aux petits éléments (Cartes, tuiles, images, videos)
- ➔ Media Queries plus adaptées à la grille et wrappers : `<body>` `<main>` `<section>` car leur taille est en général égale à celle du viewport.
- ➔ Définir leur utilisation dès la phase de conception



Des présentations à voir (ou revoir) :

- [Maîtrisez l'art du design system dans Drupal : optimisez votre style pour un thème infallible !](#)
Jeudi 11h00 . **Erwan Stephan**
- [De rôtiisseur à développeur Drupal, le parcours typique d'une reconversion.](#)
Jeudi 17h30 . **Quentin Gaudino**
- [Single Directory Component : de l'atomic design natif dans Drupal](#)
Vendredi 14h00. **Quentin Harelle**
- [Comment redonner vie à une tierce maintenance applicative \(TMA\) en état de mort cérébrale ?](#)
Vendredi 14h45 . **Nathalie Colin**





Références

- <https://css-tricks.com/say-hello-to-css-container-queries/>
- <https://www.alsacreations.com/article/lire/1915-Les-Container-Queries-en-CSS.html>
- https://www.youtube.com/watch?v=3_-Je5XpbqY Kevin Powell





Merci
pour
votre
écoute !

