

## L'elemento Picture: un'ottima soluzione per gestire le immagini responsive

La gestione delle immagini nello sviluppo di un sito web responsive è uno dei problemi attuali ancora in corso di sviluppo e dibattito. Il problema delle immagini è quello di far sì che in base al tipo di dispositivo e alla connessione utilizzata sia "servita" l'immagine giusta e meglio adatta al dispositivo che visualizza il sito web.

Quando andiamo a gestire un'immagine all'interno di un layout responsive attuiamo la seguente regola:

In questo modo l'immagine diventa fluida e si adatta alla risoluzione dello schermo del dispositivo che la visualizza. Con la regola `max-width: 100%` facciamo sì che l'immagine non superi mai la larghezza del browser.

Impostando l'altezza in "auto" l'immagine rispetterà la proporzione. Ma questa regola non basta.

Se è vero che la visualizzazione dell'immagine è perfetta e si adatta ottimamente alla risoluzione del browser, è vero anche che viene caricata dal browser nella sua grandezza originale. Per cui, per esempio, un'immagine di 1500x800 viene caricata sia nella versione desktop, sia nella versione mobile, la quale caricherà un'immagine troppo grande e conseguentemente sovraccaricherà la richiesta, determinando la lentezza del sito web.

### Esempio 1

Prova a restringere il browser: l'immagine viene visualizzata in modo corretto, ma è sempre la stessa immagine di 1688px ad essere modificata per diverse soluzioni.

## L'elemento Picture

Questa regola ci permette di visualizzare la stessa immagine sotto molte differenti circostanze, ma non ci permette di specificare diverse immagini per differenti circostanze. Questo è invece possibile grazie all'utilizzo dell'elemento Picture, il quale sfrutta una libreria JavaScript chiamata proprio `picturefill.js` che ci permette di inserire all'interno del tag diverse immagini associate a una determinata media query.

1. Per prima cosa [scarica](#) la libreria `picturefill.js` e inseriscila nel percorso del tuo sito web
2. Crea un elemento di apertura e chiusura con il tag `picture`
3. Al suo interno crea un tag `img` per ogni media query che vuoi inserire
4. Aggiungi un attributo `media` che contiene la sua caratteristica come l'altezza del viewport, la larghezza, l'orientamento, ecc.
5. Aggiungi l'attributo `srcset` con il nome dell'immagine corrispondente che vuoi caricare

6. Aggiungi un extra attributo srcset se vuoi includere differenti densità di pixel come i retina display
7. Aggiungi un elemento img in fallback

Nel mio esempio controllo se il viewport è più piccolo di 768px e in tal caso faccio caricare un'immagine più piccola rispetto a quella di default pari a 1866px; la stessa cosa se il browser ha una risoluzione pari a 480px.

### Esempio 2

Ora prova a restringere il browser. L'immagine cambia e ogni risoluzione prevede e carica un'immagine appropriata alla risoluzione del dispositivo che stai usando.

La sintassi usata con l'attributo media è lo stesso utilizzato nel CSS per creare le media query, per cui possiamo utilizzarlo anche per controllare i seguenti elementi: max-width, min-width, min-height, orientation ecc. anche per caricare le versioni panorama o il ritratto a seconda del dispositivo.

## Immagini extra per i display retina

Se vuoi fornire soluzioni differenti dell'immagini per display retina puoi aggiungere altri extra file all'attributo srcset.

### Esempio 3

In questo caso abbiamo di default 2 immagini, una per i display normali e l'altra per i display retina. Quando la risoluzione del browser è più piccola di 768px, altre due immagini di versione più piccola vengono fornite e caricate dal browser: una per i display normali e l'altra per i display retina. Viene prima verificata la media query e quindi la dimensione del browser, in questo modo viene fornita l'immagine più appropriata al tipo di display (retina o meno) e alla dimensione della risoluzione (grande e piccola).

La sintassi 1x e 2x usata con l'attributo srcset è una specie di abbreviazione che indica il tipo di densità dell'immagine: 1x per i display normali e 2x per i display retina (densità doppia).

## Compatibilità dei browser

L'immagine polyfill picture funziona su Chrome, Firefox e Opera. Se vuoi utilizzarla, ha un ottimo supporto per questi browser. Purtroppo, però, non funziona su IE9, ma puoi ovviare al problema con un accorgimento. IE9 non riconosce proprio l'elemento "source" dell'elemento picture, per questo è necessario utilizzare un codice condizionale che racchiude il codice nel tag video, così da

permettere di rendere visibili le immagini per IE9.

La soluzione con l'elemento picture, proprio perché ti permette di creare diverse combinazioni tra gli elementi quali l'orientamento, la risoluzione, la densità, oltre a essere ben integrata con gli ultimi standard del W3C, non rappresenta purtroppo una soluzione definitiva. Sicuramente ha i suoi svantaggi, uno dei quali è la impossibilità di effettuare operazioni di Art direction, cioè di specificare la parte di immagine che deve essere visualizzata dal dispositivo.

Attendiamo quindi una soluzione definitiva, ma nel frattempo vale la pena valutarla come ottima alternativa alle altre soluzioni offerte nella gestione delle immagini responsive del tuo progetto.