

スマホサイトコーディング

viewport とは

表示領域の横幅などを変更させて（仮想表示領域）
見やすい環境を提示するための記述

通常スマートフォン表示領域の横幅は960px
小さくて見づらいので最適な横幅の仮想表示領域を
指定する

viewport を記述

headタグ内にviewportを記述

```
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width,initial-scale=1">  
  <title>SITE LESSON01 </title>  
  <link rel="stylesheet" href="style.css">  
</head>
```

viewportで指定できる値

content属性内にカンマ区切りで指定する

viewport で指定できる値

width (横幅)

Viewportの横幅をピクセル単位で指定

初期値	960px
指定できる値	200px～10000px device-width : デバイス幅に合わせる

height (縦幅)

Viewportの縦幅をピクセル単位で指定（指定することはほぼない！）

初期値	横幅とのアスペクト比から計算される値
指定できる値	200px～10000px device-height : デバイス高さに合わせる

viewport で指定できる値

initial-scale (初期ズーム倍率)

初期ズーム倍率の指定

初期値	1 (width=device-width指定時)
指定できる値	数値

minimum-scale (最小倍率)

最小縮小倍率を指定

初期値	0.25
指定できる値	0～10未満の数値

maximum-scale (最大倍率)

最大拡大倍率を指定

初期値	1.6
指定できる値	0～10未満の数値

viewport で指定できる値

user-scalable (ユーザーのズーム操作の可否)

ユーザーがズームすることを許可するかどうかを指定

初期値	yes
指定できる値	yes/noまたは1/0

target-densitydpi (ターゲットとなる画面密度)

画面密度の指定 (旧版のAndroid端末向けの指定)

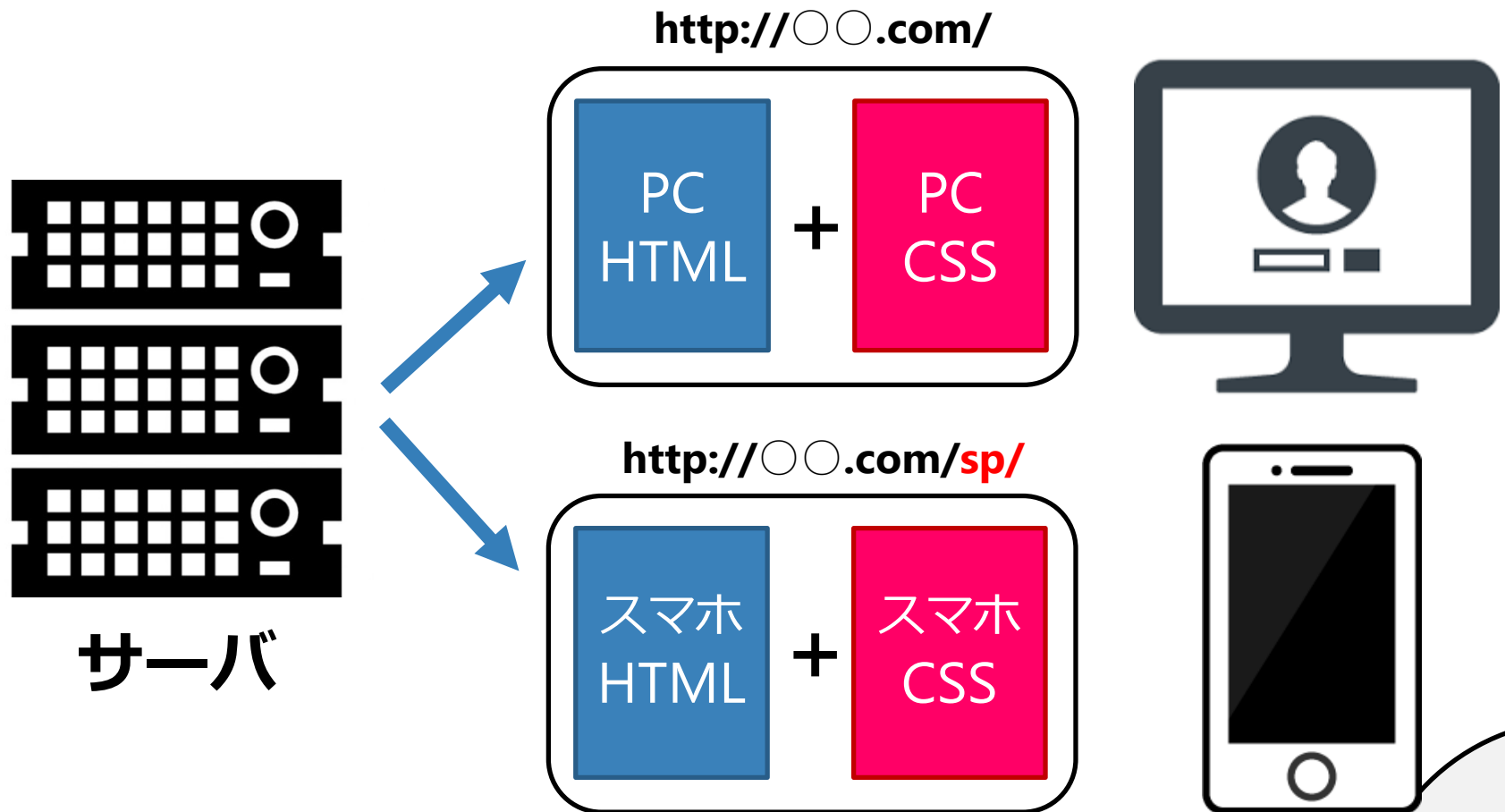
指定できる値	target-densitydpi=medium-dpiもしくはは記述しない ※最近のAndroid端末(Android4.2以降)では廃止されている
--------	--

ユーザエージェント振り分け

ユーザエージェント振り分け

ユーザエージェント

どんなデバイスからアクセスしているか判断できる情報



レスポンスイブウェブデザイン RWD

レスポンシブウェブデザイン

画面幅を判断基準にして、各デバイス（PCやスマホ等）用にページレイアウトやデザインを調整して表示させる手法

メリット

HTMLが一つですむ：文章構造の修正が1回のみ

SEO対策に有利：URLがPC・スマホ共通なのでコンテンツが分散しない
（ユーザエージェントの場合同一コンテンツが複数存在することになる）

デメリット

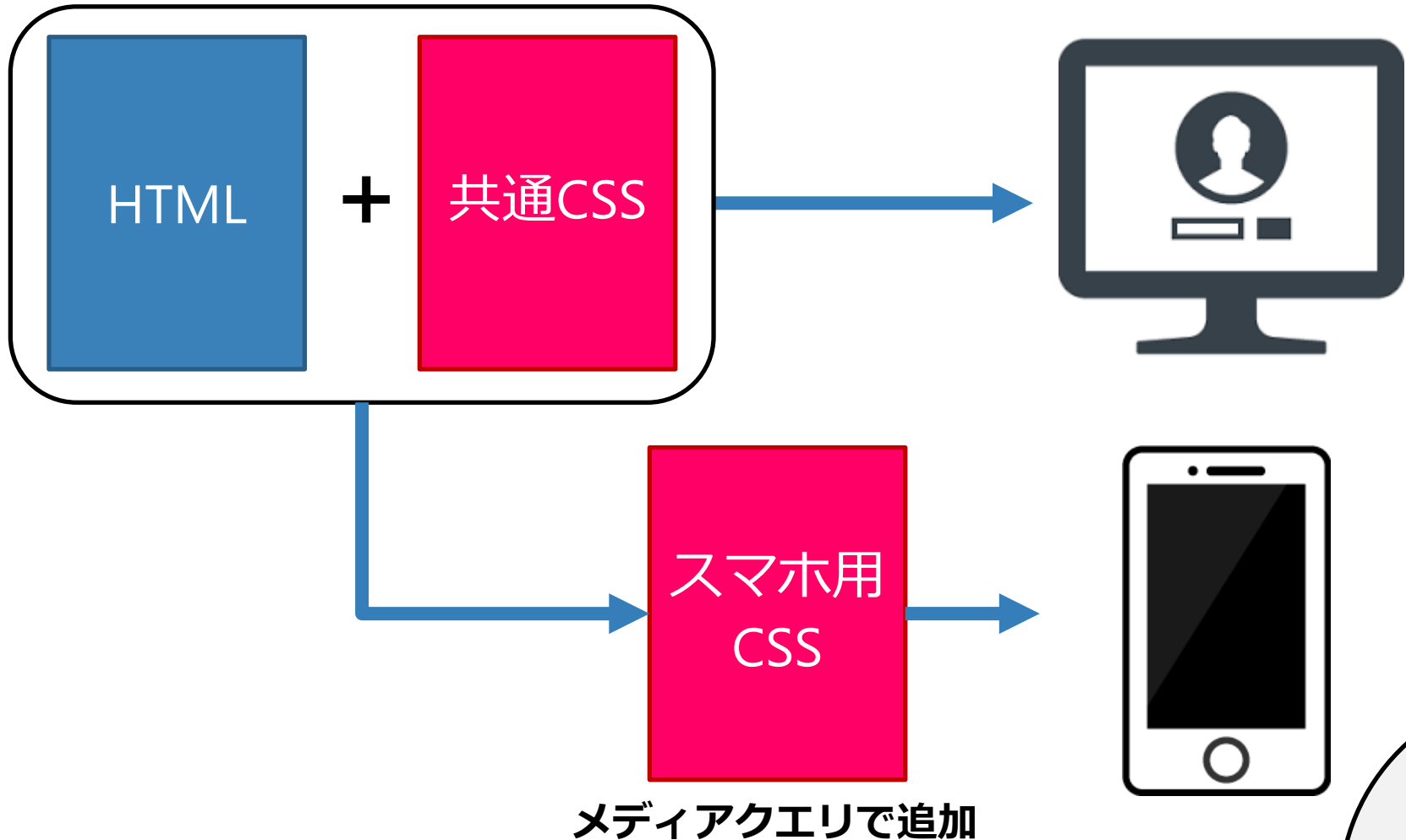
構造が複雑：各デバイスのコンテンツを一つのソースの中に記述する

基本的に同じデザイン：デバイスごとに大きくデザインを変更できない

表示が遅くなる：各デバイス用の様々なコンテンツが実は存在している

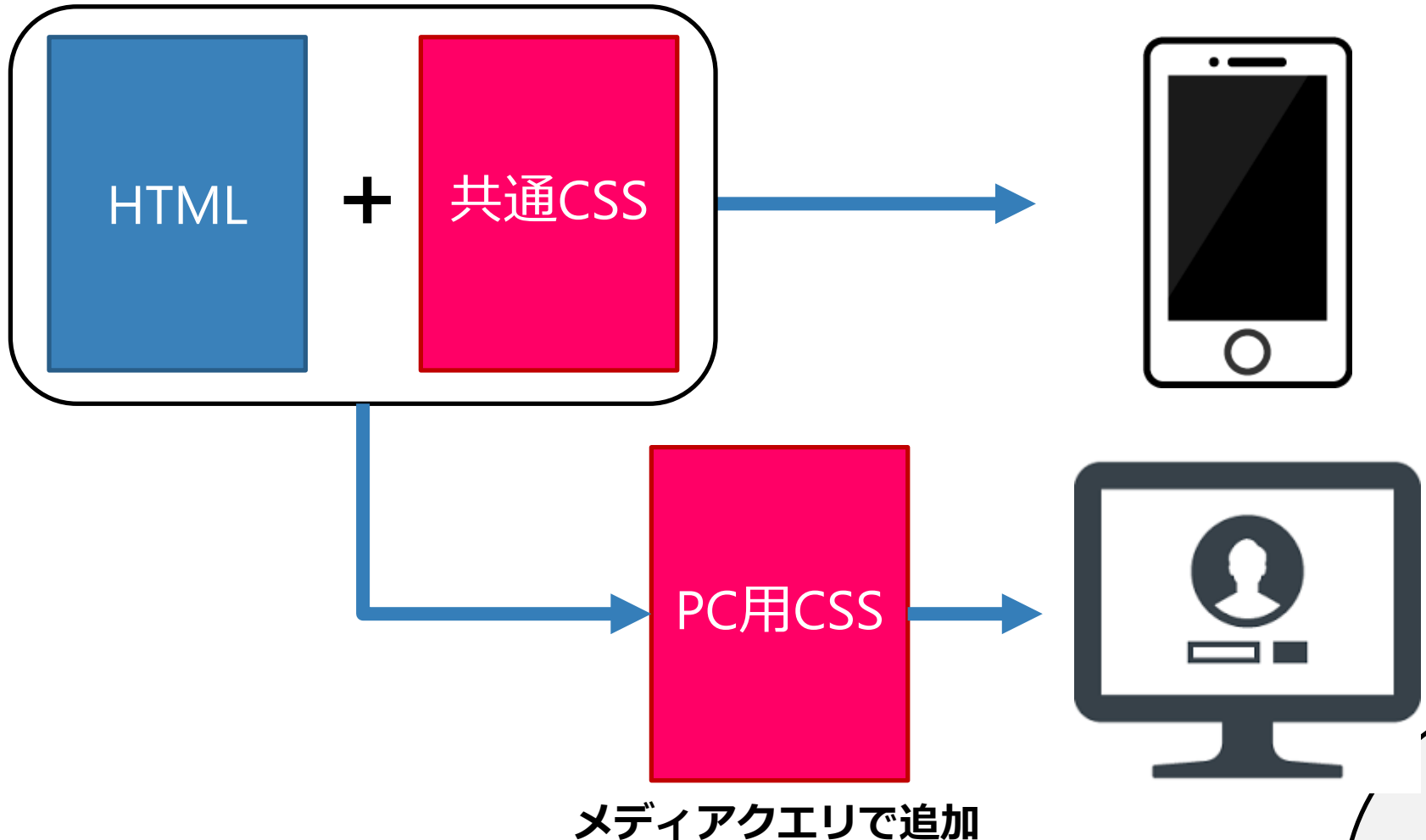
レスポンシブウェブデザイン

PCファースト：
PCデザインベースからスマホデザイン作成



レスポンシブウェブデザイン

モバイルファースト：
スマホデザインベースからPCデザイン作成



メディアクエリ

メディアクエリとは

画面幅、スマートフォンの向き、画面解像度等で別のCSSを適用させるための機能

主に画面幅に応じてCSSを切り替えることを

レスポンシブ（ウェブ）デザイン
RWD

と呼ぶ

メディアクエリで使える値

max-width

ブラウザ幅が指定した幅よりも小さい時に適用

例) max-width: 480px

ブラウザ幅が 480px 以下の時にスタイルが適用

min-width

ブラウザ幅が指定した幅よりも大きい時に適用

例) min-width: 481pxとすると、

ブラウザサイズが481px 以上の時にスタイルが適用

メディアクエリで使える値

min-height (ほとんど使わない)

ブラウザの高さが指定した高さよりも大きい時に適用

max-height (ほとんど使わない)

ブラウザの高さが指定した高さよりも小さい時に適用

orientation=portrait

ブラウザの高さが幅と同じかそれ以上の場合に適用

※スマホ縦表示時という解釈で使われることが多い

orientation=landscape

ブラウザの幅が高さよりも大きい場合に適用

※スマホ横表示時という解釈で使われることが多い

PCファースト

基本：PCデザイン → スマホデザイン

HTML

<!-- PCでもスマホでも読み込まれるCSS -->

<link rel="stylesheet" href="style.css" media="screen">

<!-- スマホ画面サイズで読み込まれるCSS -->

<link rel="stylesheet" href="mobile.css" media="screen and (max-width:480px)">

CSS

/* PCでもスマホでも適用されるデザイン定義 */

```
.contents {  
  width: 960px;  
}
```

/* スマホ用のデザイン定義 */

```
@media screen and (max-width: 480px) {  
  .content {  
    width: auto;  
  }  
}
```

モバイルファースト

基本：スマホデザイン → PCデザイン

HTML

<!-- PCでもスマホでも読み込まれるCSS -->

<link rel="stylesheet" href="style.css" media="screen">

<!-- PC画面サイズで読み込まれるCSS -->

<link rel="stylesheet" href="pc.css" media="screen and (min-width:480px)">

CSS

/* PCでもスマホでも適用されるデザイン定義 */

```
.contents {  
  width: auto;  
}
```

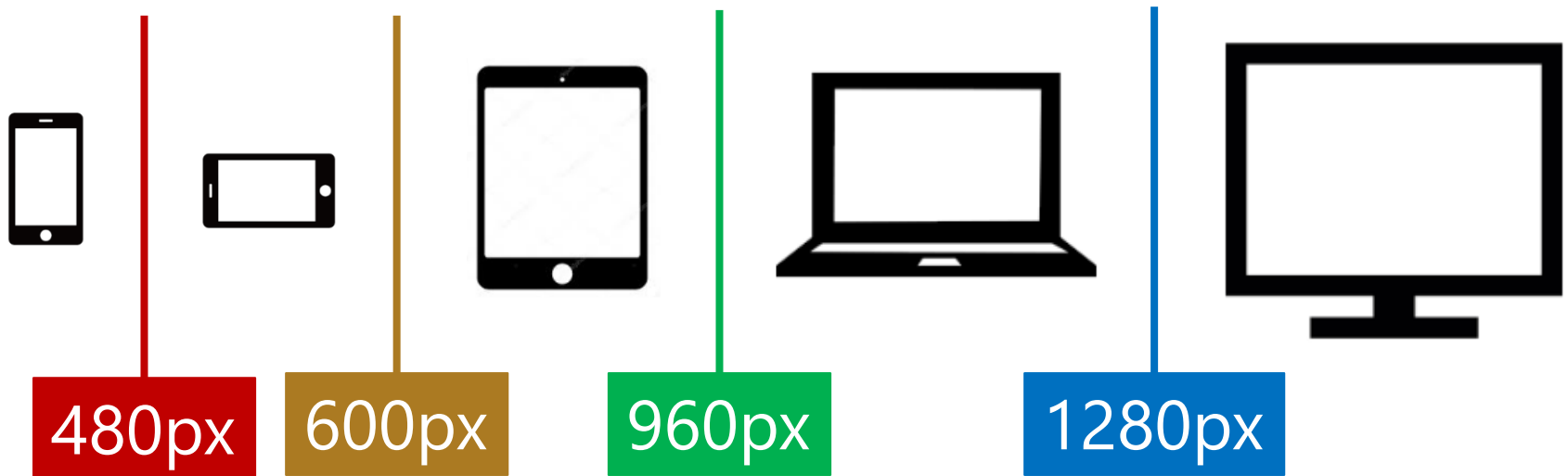
/* PC用のデザイン定義 */

```
@media screen and (min-width: 480px) {  
  .contents {  
    width: 960px;  
  }  
}
```

ブレイクポイントとは

CSSを切り替える区切りとなる画面幅

新しいスマホがでるとブレイクポイントの見直しが行われる



よく使うブレイクポイント

PCファースト

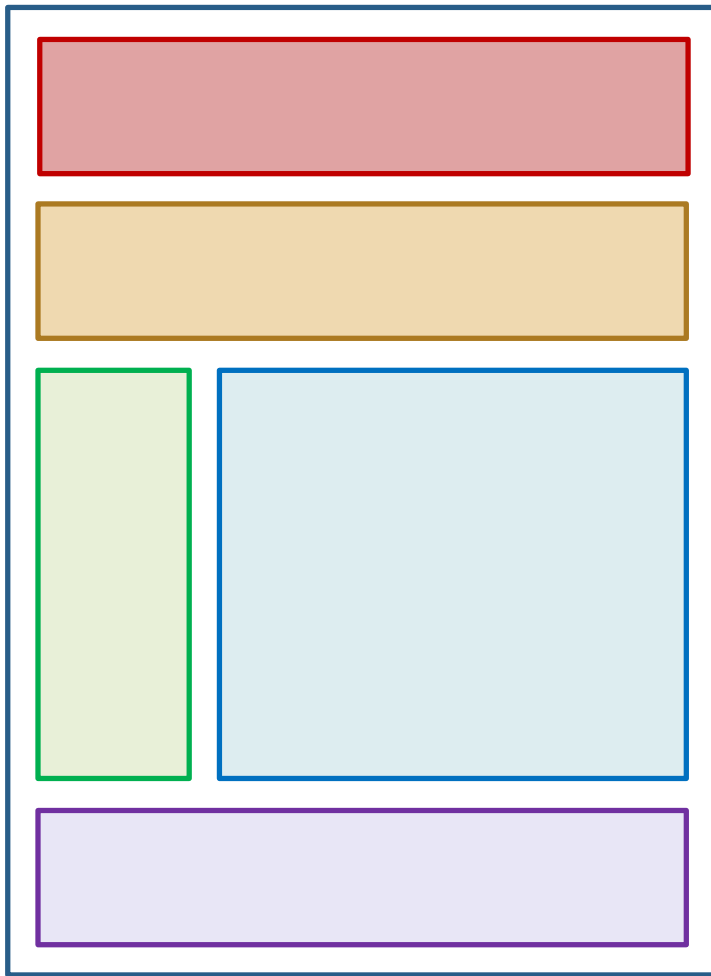
PC : 通常	-
PC : 大画面	@media screen and (min-width: 1280px)
タブレット : 縦	@media screen and (max-width: 960px)
スマホ : 横	@media screen and (max-width: 600px)
スマホ : 縦	@media screen and (max-width: 480px)

モバイルファースト

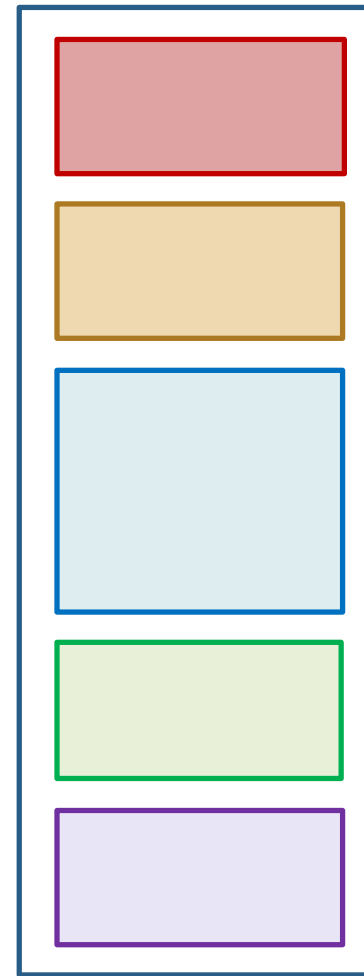
スマホ : 縦	-
スマホ : 横	@media screen and (min-width: 480px)
タブレット : 縦	@media screen and (min-width: 600px)
PC : 通常	@media screen and (min-width: 960px)
PC : 大画面	@media screen and (min-width: 1280px)

スマホサイトのCSS対応

段組みなしのレイアウトにする



PCデザイン



スマホデザイン

width指定をautoにする

```
セクタ {  
    width: auto;  
}
```

min-widthの指定を0にする

```
セクタ {  
    min-width: 0;  
}
```

画像の最大幅を100%にする

```
img {  
    max-width : 100% !important;  
    height : auto !important;  
}
```

float を解除

セクタ {

float: none;

}

display: inline-block; を解除

セクタ {

display: 元の要素の表示方法;

}

左右マージンはなるべく使用しない

セクタ {

margin-left: 0;

margin-right: 0;

}

スマホを横にした時に文字サイズが変わる対応

```
body {  
  -webkit-text-size-adjust: 100%;  
}
```

半角英数字のみURLを強制改行

```
a や .url {  
  word-break: break-all;  
}
```

テーブルの対応（divで囲って「overflow: scroll;」）

```
テーブルを囲ったdivのセレクトタ {  
  overflow: scroll;  
}
```

段組みを作成するときはflexを積極採用

親要素セレクトタ {

display: flex;

}

子要素セレクトタ {

width: 50%;

}

どうしてもはみだしてしまうものには

セレクトタ {

overflow: hidden;

}

「PCのみ表示」と「スマホのみ表示」を使い分ける

PC表示用CSS

```
.pc {  
    display: block;  
}  
.sp {  
    display: none;  
}
```

スマホ表示用CSS

```
.pc {  
    display: none;  
}  
.sp {  
    display: block;  
}
```

HTML

```
<p class="pc">PC表示のみ</p>
```

```
<p class="sp">スマホ表示のみ</p>
```

Retinaディスプレイ デバイスピクセル

人間の目ではドットを確認することができないレベルという意味で、**Retina**（網膜）ディスプレイと名付けられたらしい

iPhoneの画面サイズと解像度

サイズ	端末	CSSピクセル	Retina	デバイスピクセル
3.5	iPhone iPhone 3G, 3GS	320x480	1	320x480
3.5	iPhone 4, 4S	320x480	2	640x960
4	iPhone 5, 5s, 5c, SE	320x568	2	640x1136
4.7	iPhone 6, 6s, 7, 8	375x667	2	750x1334
5.5	iPhone 6 Plus, 6s Plus, 7 Plus, 8 Plus	414x736	3	1242x2208
5.8	iPhone X	375x812	3	1125x2436
6.1	iPhone XR	414x896	2	828x1792
6.5	iPhone XS Max	414x896	3	1242x2688

用語解説

CSSピクセル

ブラウザで表示される画面サイズ。
通常はこれを意識してプログラムを作成します。

デバイスピクセル

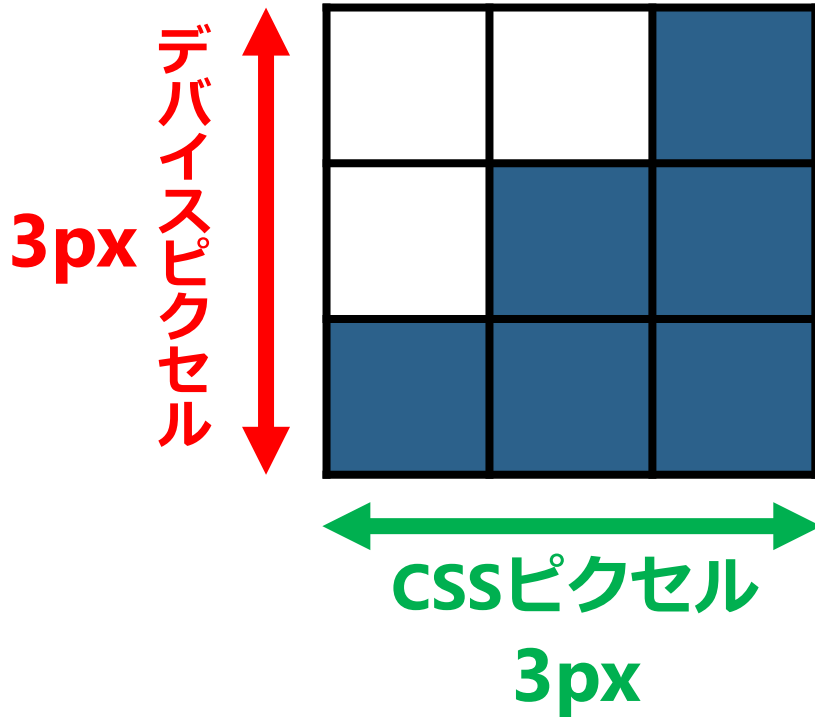
描画が行われる論理的な画面サイズ
ピクセルパーフェクトな描画が必要な場合は、
このグリッドに合うように描画を行う

デバイスピクセル比

画面内のピクセル数を増やした
高解像度ディスプレイ

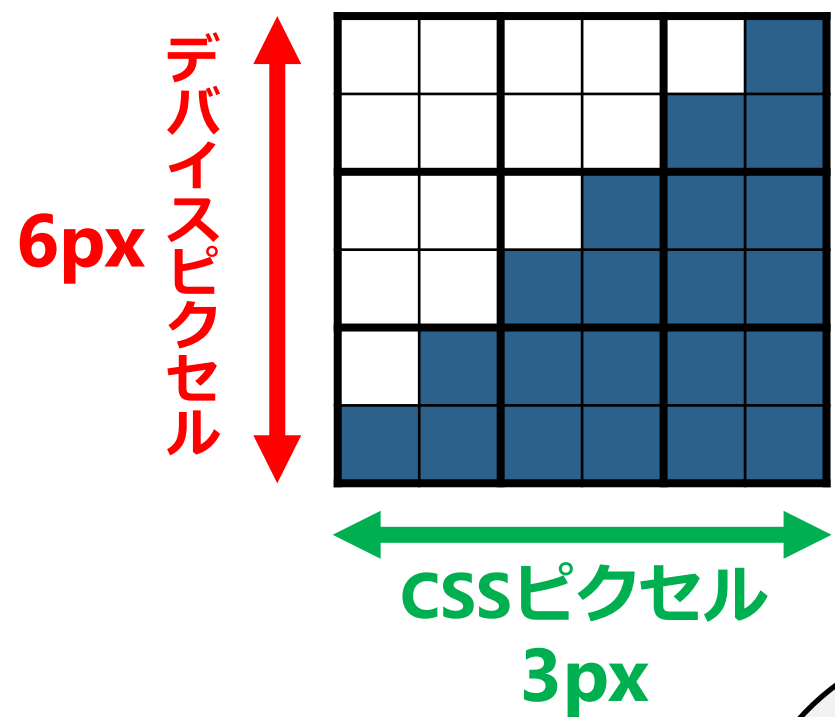
通常のディスプレイ

デバイスピクセル比 : 1



Retinaディスプレイ

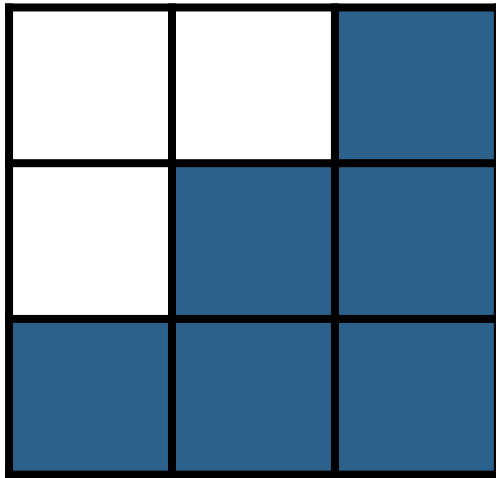
デバイスピクセル比 : 2



Retinaディスプレイで画像がぼやける

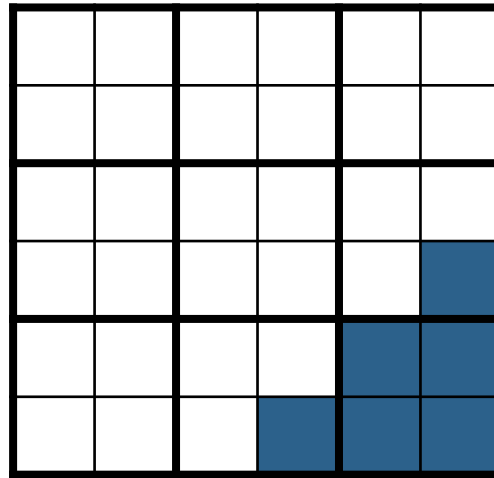
幅3px 高さ3px画像を表示

通常ディスプレイ



デバイスピクセル比 : 1

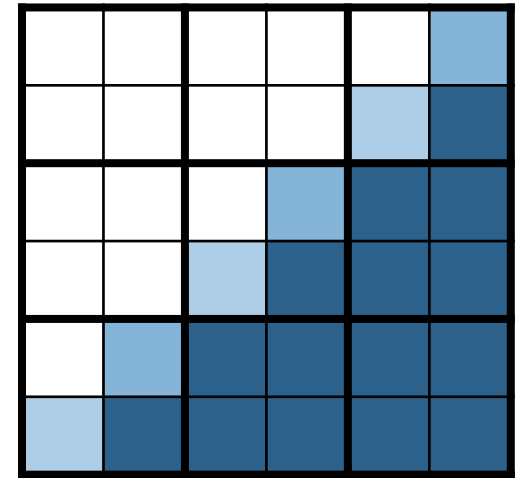
Retinaディスプレイ



デバイスピクセル比 : 1

1インチ内のピクセルが多い
画面 : 広くつかえる
大きさ : 小さくなる

Retinaディスプレイ



デバイスピクセル比 : 2

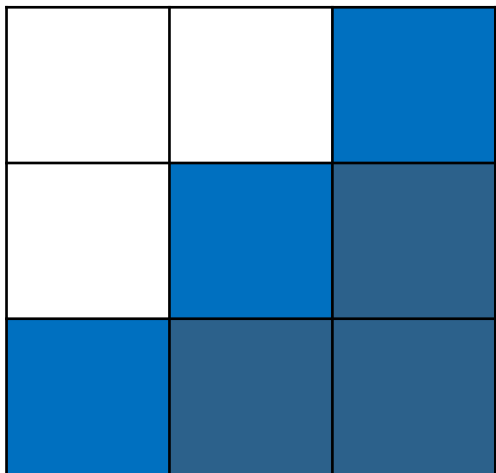
1インチ内のピクセルが多い
画面 : 通常と同じ
大きさ : 2倍に拡大して
通常と同じ大きさ

小さい画像を拡大した
ことでぼやける

画像がぼやけるの対応方法

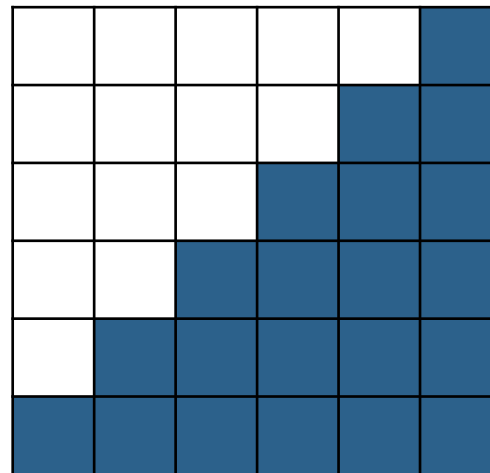
Retinaを基準にしたサイズの画像を使って表示する
幅6px 高さ6px画像を使用する

通常ディスプレイ



大きい画像を縮小して表示

Retinaディスプレイ



ピッタリサイズできれい

画像がぼやけるの対応方法

デバイスピクセル比に応じた画像を複数枚用意

picture要素

source要素を使う

img要素内でsrcset属性を使う

```
<picture>  
  <source media="(max-width: 480px)" srcset="images/retina.png">  
    
</picture>
```

```

```

IEやEdge未対応なので注意

表示領域を使った単位

表示領域（Viewport Units）を基準にした単位

※表示領域はスクロールバーも含む

Viewport Units

ブラウザーのサイズ（表示領域）が変更されるたびに
値が変わるレスポンシブな単位

vh (Viewport Height)	表示領域（viewport）の高さを基準 1vhはviewportの高さの1%に相当する
vw (Viewport Width)	表示領域（viewport）の幅を基準 1vwはviewportの幅の1%に相当
vmin (Viewport Minimum)	表示領域（viewport）の（高さと幅のうち） 小さいほうの値 を基準 viewportの高さが幅より小さい場合、1vminはviewportの高さの1%に相当 同様にviewportの幅が高さより小さい場合、1vminはviewportの幅の1%に相当
vmax (Viewport Maximum)	表示領域（viewport）の（高さと幅のうち） 大きいほうの値 を基準 viewportの高さが幅より大きい場合、1vmaxはviewportの高さの1%に相当 同様にviewportの幅が高さより大きい場合、1vmaxはviewportの幅の1%に相当

Viewport Units

ファーストビュー高さいっぱいコンテンツを表示させる
文字サイズ指定には不向き
幅指定にも不向き（スクロールバーを含んだ幅指定の為）

vh (Viewport Height)	表示領域（viewport）の高さに対する割合 1vhはviewportの高さの1%に相当
vw (Viewport Width)	表示領域（viewport）の幅に対する割合 1vwはviewportの幅の1%に相当
vmin (Viewport Minimum) IE9はvmを使用	表示領域（viewport）の幅と高さのうち 小さいほうの値 に対する割合 viewportの幅が高さより小さい場合 1vminはviewportの幅の1%に相当 同様にviewportの高さが幅より小さい場合 1vminはviewportの高さの1%に相当
vmax (Viewport Maximum) IE未対応	表示領域（viewport）の幅と高さのうち 大きいほうの値 に対する割合 viewportの幅が高さより大きい場合 1vmaxはviewportの幅の1%に相当 同様にviewportの高さが幅より大きい場合 1vmaxはviewportの高さの1%に相当