



过渡 transtion

讲师：许井龙

微信：ngsteel

一. 过渡简介

二. 过渡属性

1. 具有过渡效果的CSS样式属性
2. 过渡持续的时间
3. 过渡延迟时间
4. 过渡效果
 - 关键字
 - 三次贝塞尔函数及贝塞尔曲线

三. 过渡简写及过渡覆盖问题

四. 单向过渡VS双向过渡

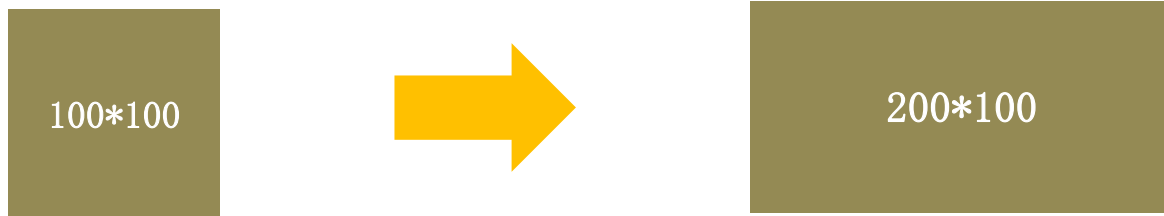
五. 过渡触发

六. 隐式过渡

七. 事件：过渡结束

八. 练习：过渡的综合运用

- 什么是过渡 (transition) ?
 - 过渡(transtion)是CSS3新增的一个样式属性。得益于CSS3提供过渡属性，让我们不再依赖JS或者flash来实现动画效果。



需求说明:

- 当鼠标悬浮在棕色背景的矩形时。
- 仅让其宽度变成原来的2倍。
- 变化的过程持续3秒钟。

1.设置过渡触发的条件

*:hover*状态伪类

2.指定过渡效果的CSS样式

width

3.设置过渡的持续时间

3s

这也是实际开发中，实现过渡效果的关键步骤！

- 默认情况，浏览器会为每个元素设置，
 - transition-property: all
 - 参数all表示所有具有中间值的CSS样式属性名，**只要样式有值的变化，且设置了过渡时间**，就会发生过渡效果。
- 除非我们有以下需求，才需要特殊设置
 1. **仅元素的宽度有过渡效果（此时即便有其他CSS样式属性值发生变化，也不发生过渡动画效果）**
 1. transition-property: width
 2. **仅宽度和字体颜色有过渡效果（注意：多个CSS样式使用逗号分隔）**
 - transition-property: width , color
- 注意：类似于 visibility的CSS样式，无论如何也不会有过渡效果，**因为visibility的值即不是数值，也无法转为数值。**
 - transition-property: visibility

- transition-duration
 - **该属性是过渡的核心**，设置过渡效果持续的时间。单位秒(s)或者毫秒(ms)
 - **默认 0s**，即没有过渡效果（如果没有指定过渡时间，是无论如何也无法产生过渡效果的）
- 过渡样式属性的可设置的值
 - transition-duration: .3s 过渡持续时间 300毫秒，推荐写法
 - transition-duration: 0.1s 过渡持续时间 100毫秒，**不推荐该写法**
 - transition-duration: 200ms 过渡持续时间 200毫秒
 - transition-duration: 2s 过渡持续时间 2秒
 - transition-duration: 2s, .5s **多个过渡时间使用逗号分隔，通常配合控制多个过渡属性。**

多个过渡时间，控制多个属性

transition-property: pro1, pro2, pro3, pro4, pro5, pro6,
pro7

transition-duration: 2s

所有属性的过渡时间都是2s

transition-property: pro1, pro2, pro3, pro4, pro5, pro6,
pro7

transition-duration: 2s, .3s

过渡时间为 2s的属性是 pro1 pro3 pro5 pro7

过渡时间为 .3s的属性是 pro2 pro4 pro6

transition-property: pro1, pro2, pro3, pro4, pro5, pro6, pro7

transition-duration: .2s, 5s, .1s

过渡时间为 .2s的属性是 pro1 pro4 pro7

过渡时间为 5s的属性是 pro2 pro5

过渡时间为 .1s的属性是 pro3 pro6

- transition-delay
 - 设置过渡延迟执行时间。
 - 单位秒或者毫秒。
 - 默认 0s，即立即执行过渡效果。可以负值。
- 过渡的参考值
 - transition-delay : .3s 过渡延迟300毫秒后执行，推荐写法
 - transition-delay : 0.1s 过渡延迟100毫秒后执行，**不推荐该写法**
 - transition-delay : 200ms 过渡延迟200毫秒后执行
 - transition-delay : 2s 过渡延迟2秒后执行
 - transition-delay : -1s **如果过渡持续时间3秒,实际看到的过渡效果只有后2秒(3s - 1s), 前1秒会立即执行, 并没有任何过渡效果。**
 - transition-delay : 2s, .5s 多个过渡时间使用逗号分隔，通常配合控制多个过渡属性。

多个过渡延迟时间

transition-property: pro1, pro2, pro3, pro4, pro5, pro6, pro7

transition-delay: 2s

所有属性的过渡动画都会延时2秒执行

transition-property: pro1, pro2, pro3, pro4, pro5, pro6, pro7

transition-delay: 2s, .3s

过渡动画都会延时 2s 的属性是 pro1 pro3 pro5 pro7

过渡动画都会延时 .3s 的属性是 pro2 pro4 pro6

transition-property: pro1, pro2, pro3, pro4, pro5, pro6, pro7

transition-delay: .2s, 5s, .1s

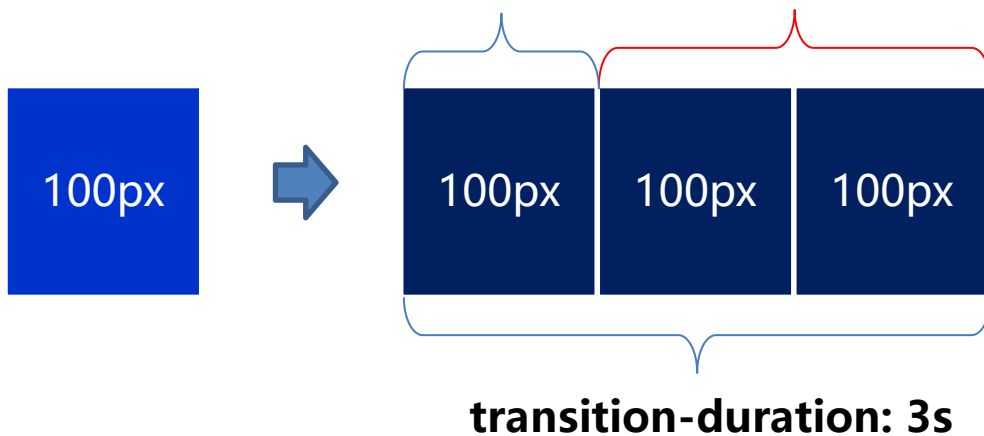
过渡动画都会延时 .2s 的属性是 pro1 pro4 pro7

过渡动画都会延时 5s 的属性是 pro2 pro5

过渡动画都会延时 .1s 的属性是 pro3 pro6

transition-delay: -1s;

只有后2秒有过渡效果



此时我们看到的过渡动画持续时间2秒

- 控制在整个过渡动画时间内不同时间段内变换的快慢，以及值的大小。CSS提供了5关键字，默认
 - transition-timing-function: ease

值	描述
linear	规定以相同速度开始至结束的过渡效果（等于 cubic-bezier(0,0,1,1)）。
ease	规定慢速开始，然后变快，然后慢速结束的过渡效果（cubic-bezier(0.25,0.1,0.25,1)）。
ease-in	规定以慢速开始的过渡效果（等于 cubic-bezier(0.42,0,1,1)）。
ease-out	规定以慢速结束的过渡效果（等于 cubic-bezier(0,0,0.58,1)）。
ease-in-out	规定以慢速开始和结束的过渡效果（等于 cubic-bezier(0.42,0,0.58,1)）。
cubic-bezier(n,n,n,n)	在 cubic-bezier 函数中定义自己的值。可能的值是 0 至 1 之间的数值。

- 所有关键字都可以转换为cubic-bezier（），在实际开发中更倾向使用cubic-bezier（），我们可以通过在线工具生成该函数。

cubic-bezier(0,1.5,1,-0.38)

cubic-bezier(0,1.5,1,-0.38) SAVE

This curve contains values out of range. But fear not young padawan! Just use `cubic-bezier(0,1,1,0)` as well for Webkit until the [bug #45761](#) fix propagates to Safari.

Preview & compare GO!

Duration: 1 second

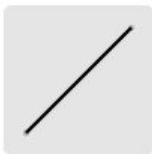


Library IMPORT EXPORT

Click on a curve to compare it with the current one.



ease



linear



ease-in



ease-out



ease-in-out

Tip: Right click on any library curve and select "Copy Link Address" to get a permalink to it which you can share with others

CSS样式值的
变换曲线

过渡时间

cubic-bezier(1,1.3,0,-0.38)

cubic-bezier(1,1.3,0,-0.38) SAVE

This curve contains values out of range. But fear not young padawan! Just use `cubic-bezier(1,1,0,0)` as well for Webkit until the [bug #45761](#) fix propagates to Safari.

Preview & compare GO!

Duration: 1 second



Library IMPORT EXPORT

Click on a curve to compare it with the current one.



ease



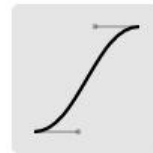
linear



ease-in



ease-out



ease-in-out

Tip: Right click on any library curve and select "Copy Link Address" to get a permalink to it which you can share with others

cubic-bezier(.66,-0.53,.41,1.52)

cubic-bezier(.66, -0.53, .41, 1.52) SAVE

This curve contains values out of range. But fear not young padawan! Just use `cubic-bezier(.66,0,.41,1)` as well for Webkit until the [bug #45761](#) fix propagates to Safari.

Preview & compare GO!

Duration: 1 second

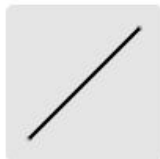


Library IMPORT EXPORT

Click on a curve to compare it with the current one.



ease



linear



ease-in



ease-out



ease-in-out

Tip: Right click on any library curve and select "Copy Link Address" to get a permalink to it which you can share with others

- 在实际开发过程中，我们推荐使用简写过渡。其语法如下，
 - ① 最简单的方式，只需要设置过渡时间。
 - ◆ `transition: 过渡时间`
 - ② 标准简写方式。
 - ◆ `transition: CSS属性名1 过渡时间 过渡效果 过渡延迟时间`
 - ③ 多个属性控制，使用逗号分割简写方。
 - ◆ `transition: CSS属性名1 过渡时间1 过渡效果1 过渡延迟时间1, CSS属性名2 过渡时间2 过渡效果2 过渡延迟时间 2`
- 特别说明
 - 当我们不显示指定CSS样式属性，浏览器默认为all，让所有样式属性的值是数值类型的CSS样式具有过渡效果。

/*

默认: 逗号后的相同CSS名设置的过渡样式, 会覆盖前面的过渡样式

*/

```
transition:width 10s, all 2s -.5s;
```

/*

特殊情况: 逗号后的相同CSS名设置的过渡样式, 过渡时间为零, 使用前面的过渡样式。

过渡时间为零的两种情况

- 1、使用负延迟时间抵消 例如 transition:width 3s, all 1s -1s;
- 2、显示设置0 例如 transition:width 3s, all 0s;

*/

```
transition:width 3s, all 0s;
```


- **当通过“状态伪类”触发过渡**

- ✓ 过渡样式设置在“样式开始值”位置（触发过渡，状态消失都会产生过渡效果）
- ✓ 过渡样式设置在“样式结束值”位置（只有触发过渡时，会产生过渡效果）

单向过渡VS双向过渡

```
.box {
```

```
width: 100px;
```

```
height: 100px;
```

```
background-color: #00f;
```

```
}
```

开 始

样 式

```
.box {
```

```
width: 100px;
```

```
height: 100px;
```

```
background-color: #00f;
```

```
transition: 3s;
```

```
}
```

单向过渡



VS



双向过渡

```
body:hover .box {
```

```
width: 400px;
```

```
transition: 3s;
```

```
}
```

结 束

样 式

```
body:hover .box {
```

```
width: 400px;
```

```
}
```

- 状态伪类
 - :hover
 - :active
 - :focus
- JS
 - 直接添加样式
- @media
 - 在不通过的视口显示不同的样式

- Firefox 与 Chrome 会触发隐式过渡。

```
.box {  
    width: 5em;  
    height: 5em;  
    background-color: #00f;  
    color: #fff;  
    font-size: 20px;  
}  
  
body:hover .box {  
    font-size: 40px;  
    transition: font-size 3s;  
}
```

上述代码中，仅指定了字体大小的过渡，但是在firefox 导致了宽高的“隐式过渡”

- 过渡结束后，触发transitionend事件。

```
var element = document.getElementById("slidingMenu");  
element.addEventListener("transitionend", function(event) {  
    element.innerHTML = "过渡时间结束!";  
}, false);
```

事件属性	类型	描述
target	EventTarget	触发过渡结束事件的源
type	DOMString	事件的类型
bubbles	Boolean	是否支持事件冒泡
cancelable	Boolean	是否取消事件冒泡
propertyName	DOMString	与过渡关联的CSS属性名
elapsedTime	Float	过渡的持续时间 (如果延迟时间为正数,含延迟时间)

重点总结

1. 设置过渡的持续持续时间

- a. 默认值为 0s，即默认没有过渡
- b. 只要该值大于 0s 即可。

2. 设置过渡的触发条件

- a. 状态伪类触发
- b. JS触发
- c. @media触发

1. 当通过状态伪类触发过渡

- ✓ 过渡样式设置在“样式开始值”位置（触发过渡，状态消失都会产生过渡效果）
- ✓ 过渡样式设置在“样式结束值”位置（只有触发过渡时，会产生过渡效果）

2. 过渡简写导致样式覆盖问题

- ✓ 逗号后面的样式会覆盖前面的样式。

- 缓慢缩放
 - 参考电商商品放大效果
- 缓慢位移
 - 参考移动端主菜单
- 淡入淡出
- 碰撞效果
- 旋转效果 (配合 transform)

