Hello.

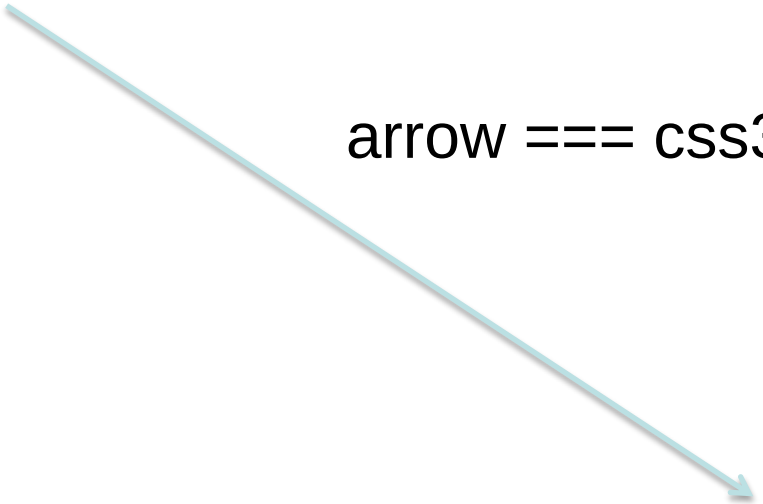
CSS Transitions, Transforms &Animations
Tuesday February 23, 2011

HUGE
45 Main Street, 2nd Floor, Brooklyn, NY 11201
718 625 4843—www.hugeinc.com

transitions

Ch-ch-ch-ch-changes

Class .a



arrow === css3 transition == ZOMG

Class .b

What (are CSS3 transitions)?

Sometimes we change CSS properties dynamically by changing or adding a class.
e.g. $('input').addClass('error');

CSS3 Transitions allow us to animate the change in these CSS properties… easily!

Transitions defined & browser support

-browser-transition: {property} {duration} {easing} {delay};

easing defaults: linear, ease-in, ease-out, ease-in-out

-webkit-transition: (Safari 3.1+, Chrome since forever)
-moz-transition: (FF 4+)
-o-transition: (Opera 10.doYouCare)
-msie-transition: (apparently NOT ie9 ☹)

animations

keyframes

```
from {
left: 100px;
width: 100px;
height: 100px;
animation-timing: ease-in;
}
```

```
to {
left: 200px;
width: 50px;
height: 50px;
}
```

but what's going on in the middle?

changing multiple styles at different times over an interval

keyframes (cont.)

```
from {
...
}
```

```
30% {
left: 200px;
width: 50px;
height: 50px;
}
```

```
60% {
left: 100px;
}
```

```
to {
transform: rotate(180deg);
left: 200px;
}
```

using animations programmatically

Initializer

- initAnimationEvent - Initializes an animation event given a DOMobject
  - typeArg (string)
  - canBubbleArg (boolean)
  - cancelableArg (boolean)
  - animationNameArg (string)
  - elapsedTimeArg (float)

Callbacks

- animationStart
- animationEnd
- animationIteration

options

animation-delay: 5s;

- delay to start

animation-direction: alternate;

- animation is played in reverse on odd iterations

animation-durations: 5s;

- time to complete animation

animation-iteration-count: 5;

- times to play animation (doubled for alternate)

animation-name: myAnimation;

- unique ID for animation

animation-play-state: paused;

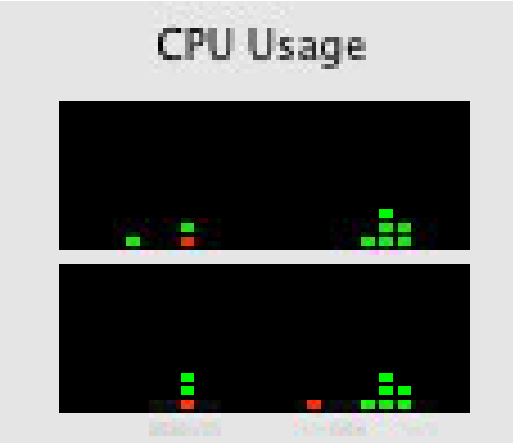
- pauses/plays the animation

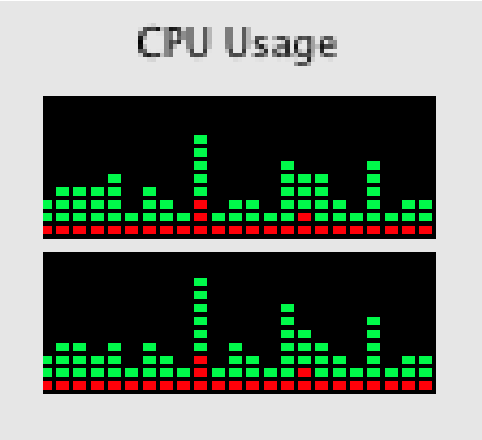
animation-timing-function: cubic-bezier(x1,y1,x2,y2)

- a custom/predefined timing curve to follow

GPU vs. CPU
hardware acceleration

It’s the difference between...





GPU’s are very good BitBLIT Operations

into the mainstream

Banner Ads

Purely Native Web Apps

Native-Like Interfaces

Immersive Sites

developers don’t want to code animations

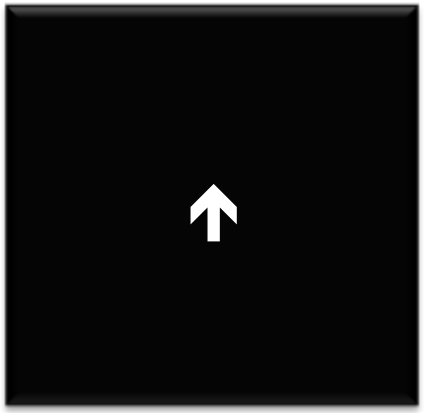
Animation Builders (Sencha Animator)

examples

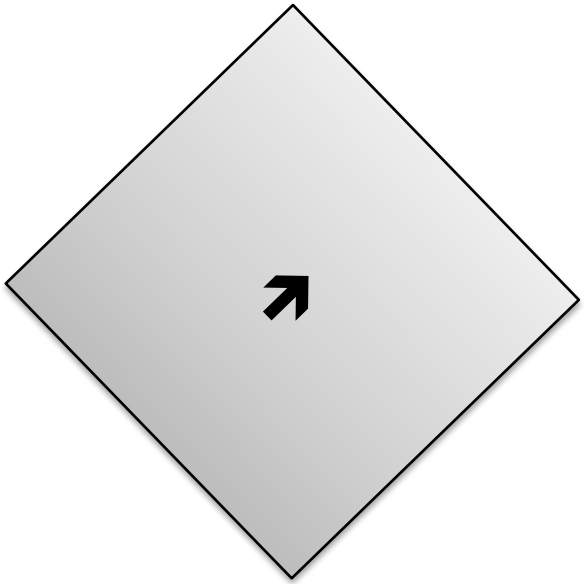
- OSX Dock
- 404 Page
- Solar System
- Portfolio

transforms

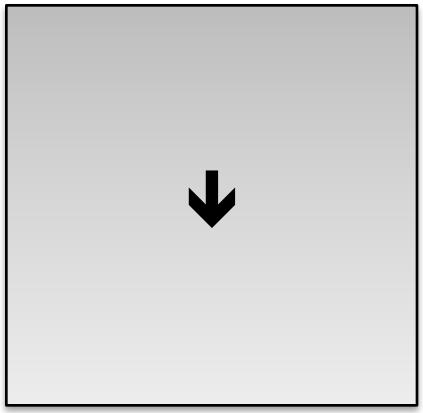
transform: rotate(x)



rotate(0deg)

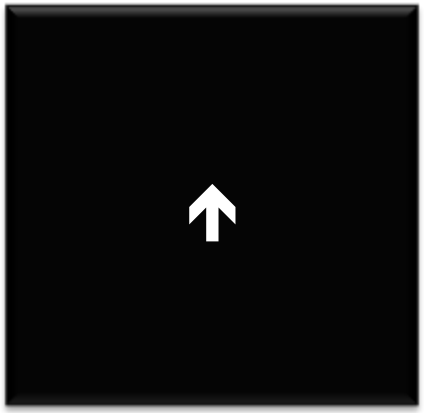


rotate(45deg)

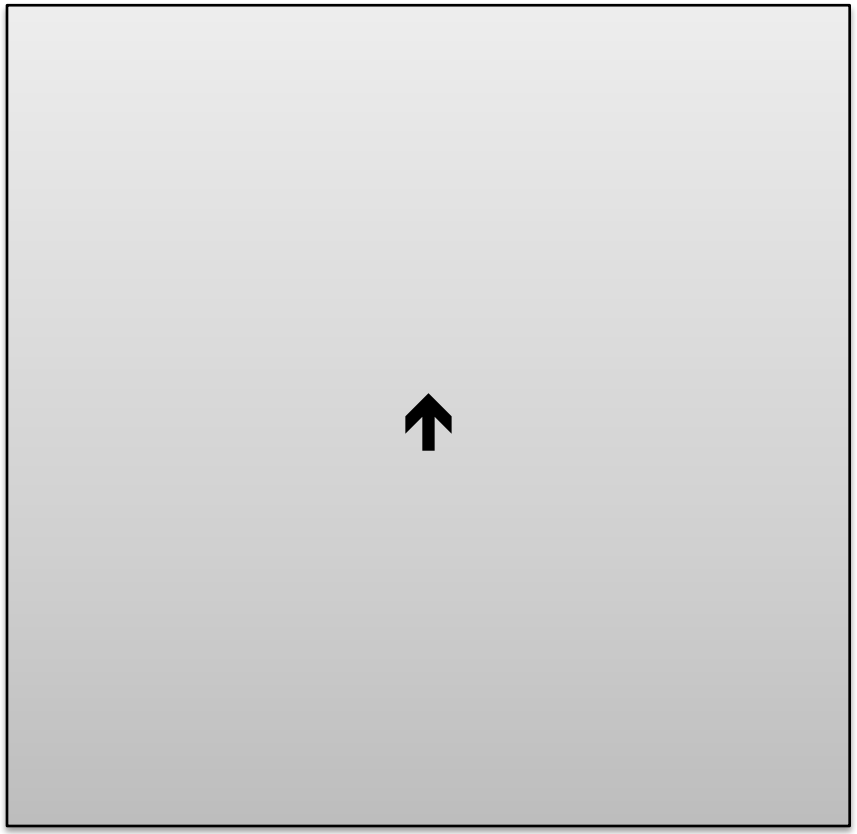


rotate(180deg)

transform: scale(x)

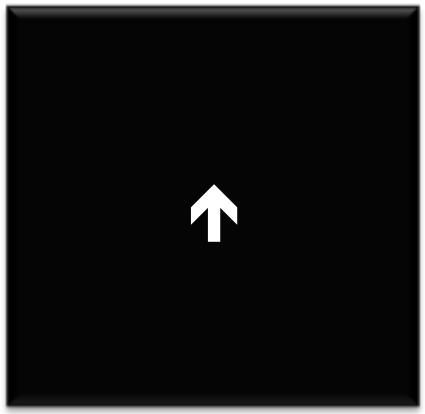


scale(1)

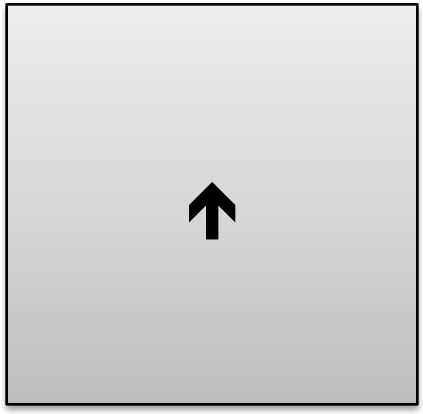


scale(2)

transform: translate(x, y)

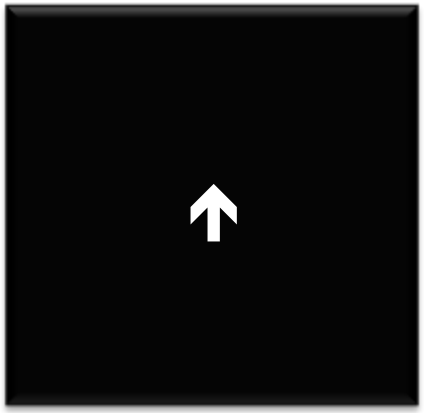


translate(0,0)



translate(200px, 100px)

transform: skew(x, y)

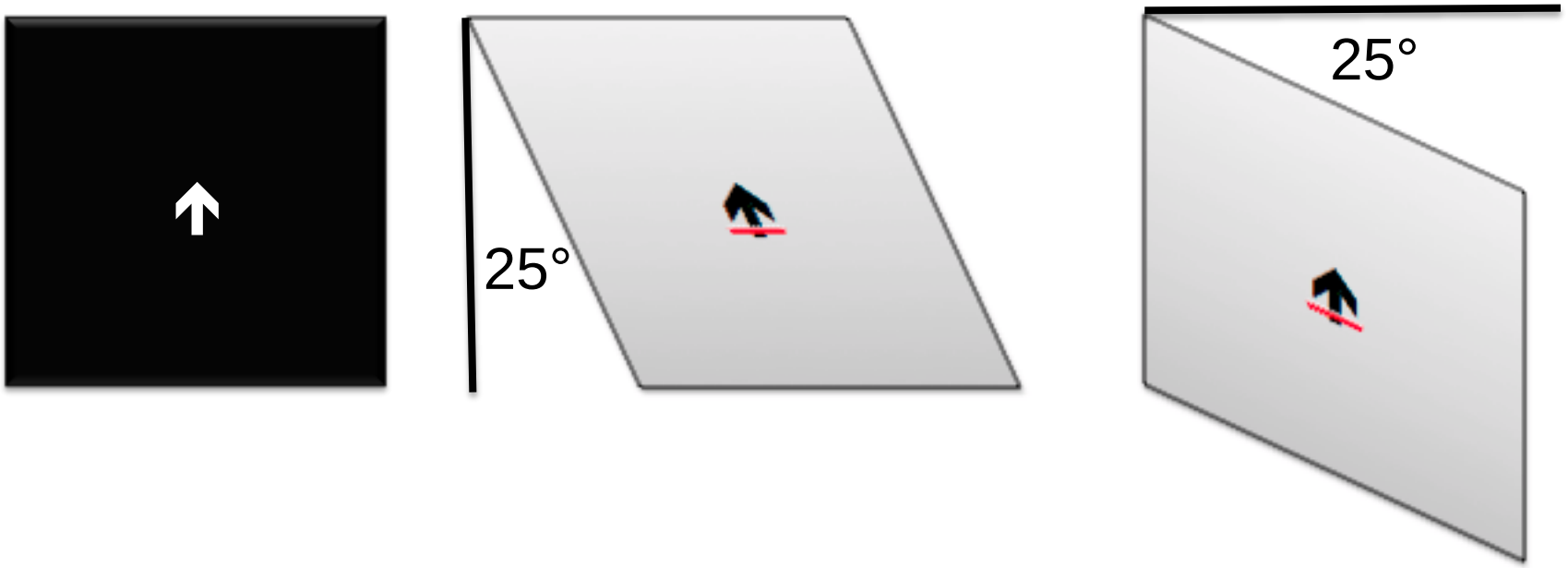


skew(0deg, 0deg)

skewX(25deg)

skewY(25deg)

transform: skew(x, y)

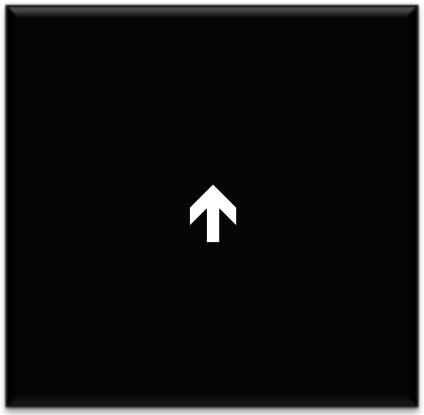


skew(0deg, 0deg)

skewX(25deg)

skewY(25deg)

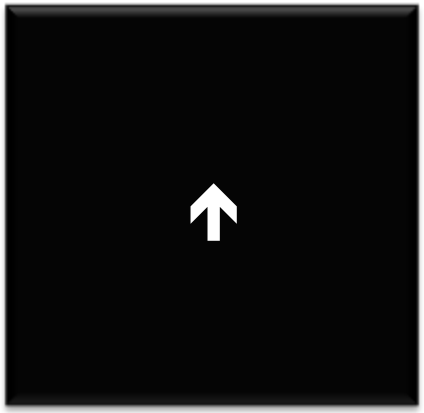
transform: skew(x, y)



skew(0deg, 0deg)

skew(25deg, 25deg)

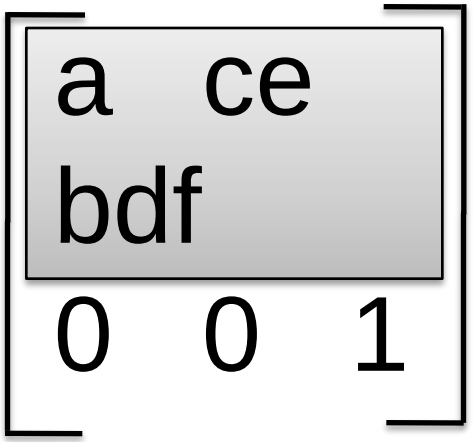
transform: matrix(a,b,c,d,e,f)



???

transform: matrix(a,b,c,d,e,f)

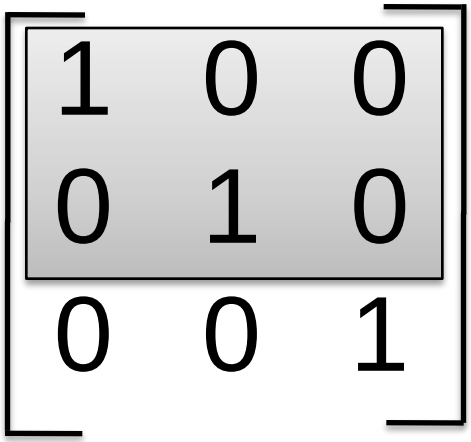
matrix(a,b,c,d,e,f) =



3x3 matrix

transform: matrix(a,b,c,d,e,f)

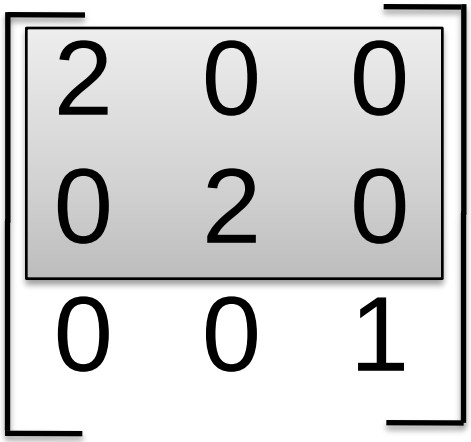
matrix(1,0,0,1,0,0) =



3x3 matrix

transform: matrix(a,b,c,d,e,f)

matrix(2,0,0,2,0,0) =



3x3 matrix

transform: matrix(a,b,c,d,e,f)

matrix(2,0,0,2,0,0) = scale(2)

transform: matrix(a,b,c,d,e,f)

$$\begin{bmatrix} sx & 0 & 0 \\ 0 & sy & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

scale

$$\begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix}$$

translate

$$\begin{bmatrix} \cos(a) & -\sin(a) & 0 \\ \sin(a) & \cos(a) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

rotate

$$\begin{bmatrix} 1 & \tan(a) & 0 \\ \tan(a) & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

skew

transform: matrix(a,b,c,d,e,f)

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

flip horizontal

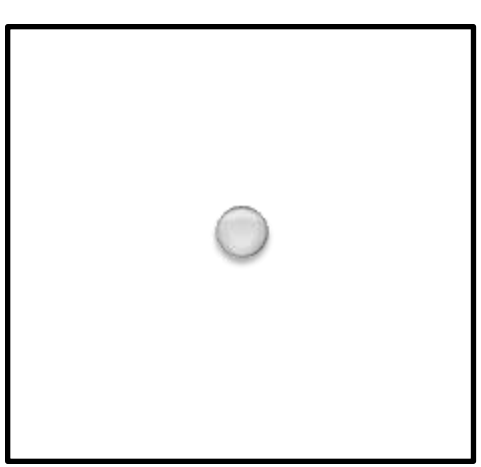
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

flip vertical

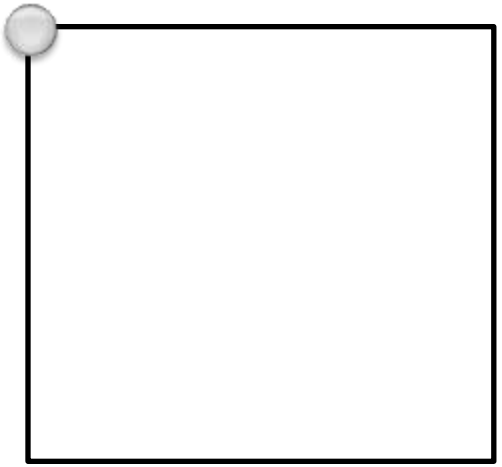
$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

flip both

transform-origin: x, y



transform-origin: 50% 50%



transform-origin: top left

Browser Support

```
-moz-transform: rotate(45deg);        // FF3.5+
-webkit-transform: rotate(45deg);     // Saf3.1+, Chrome
-o-transform: rotate(45deg);          // Opera 10.5
-ms-transform: rotate(45deg);         // IE9
transform: rotate(45deg);

filter: progid:DXImageTransform.Microsoft.Matrix(
    M11=0.7071067811865476,
    M12=-0.7071067811865476,
    M21=0.7071067811865476,
    M22=0.7071067811865476,
sizingMethod='auto expand');          // IE6 – IE9
```

3D Transformations

What's under the hood?

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

4x4 matrix

3D functions

- rotate3d
- translate3d
- skew3d
- scale3d
- matrix3d

Other properties

- transform-origin
- transform-style
- perspective
- perspective-origin
- backface-visibility

Examples

- Snow Stack (safari only)
- Morphing Cube (safari only)
- Photo Cube (safari only)
- Bezier Builder (webkit only)