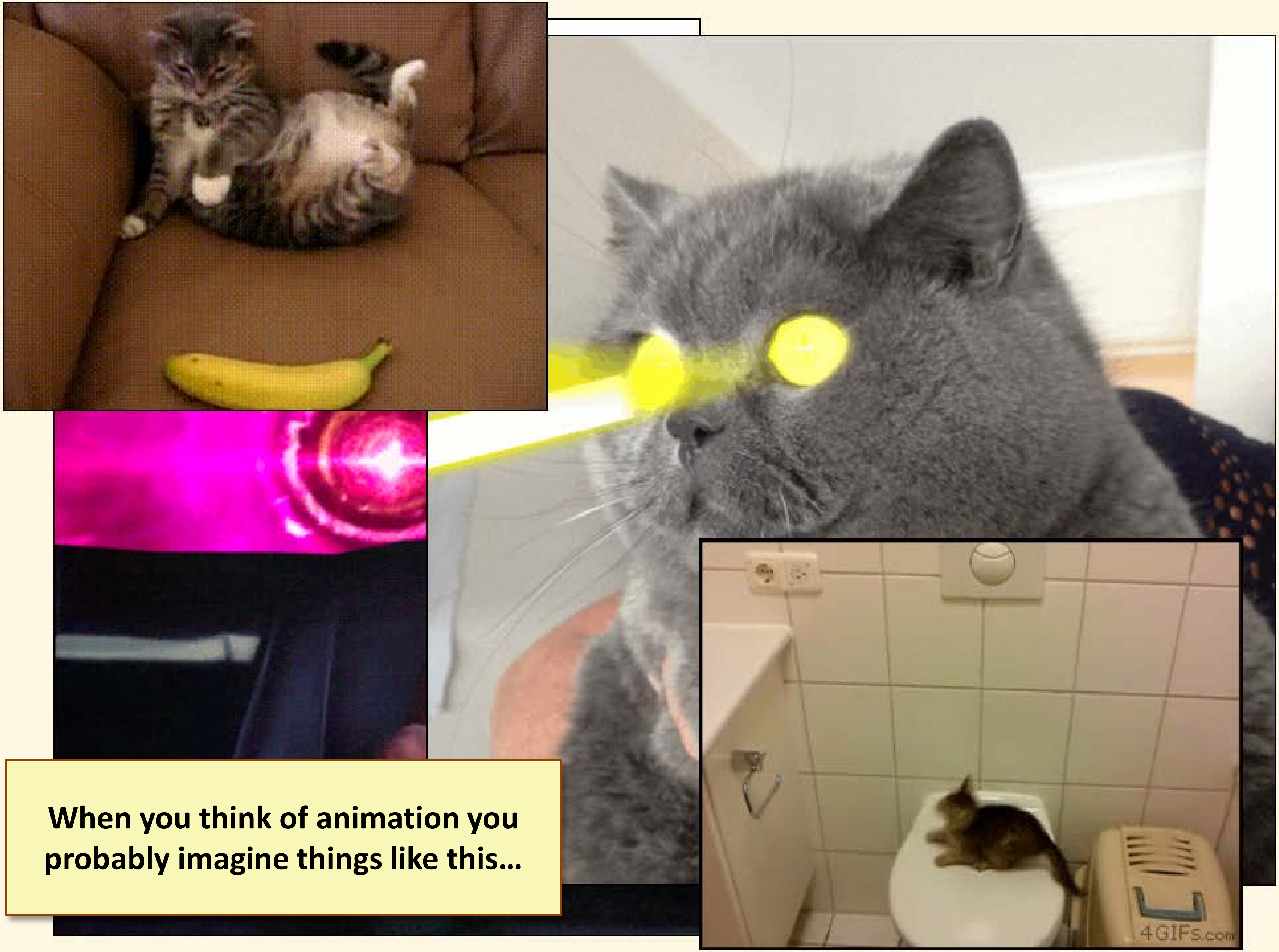
Serious animation

html5j 2013, Tokyo

Brian Birtles, Mozilla Japan

Topics

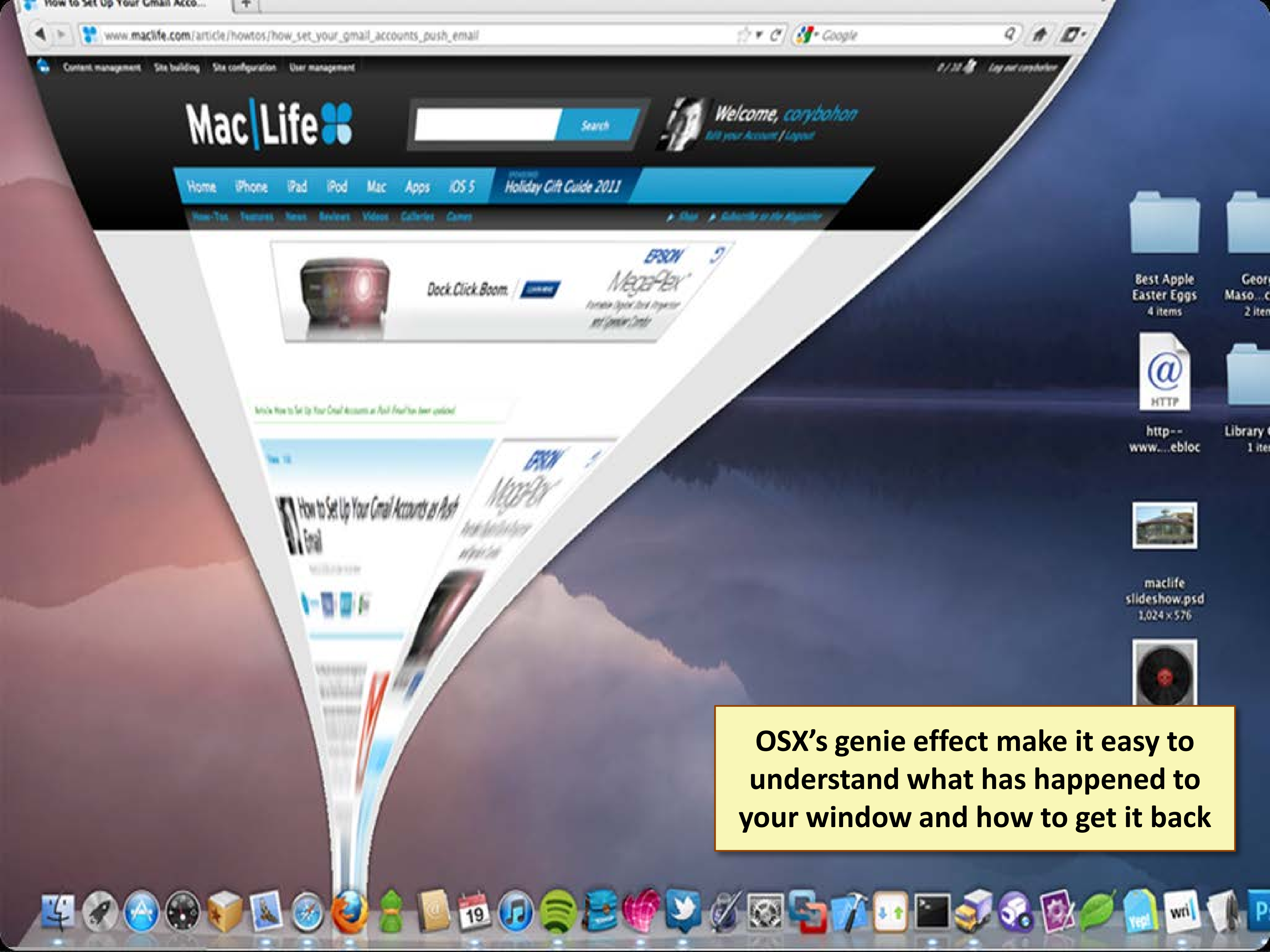
- **Serious animation**
  There's more than cat gifs?
- **Difficult animation**
  Problems with the current Web platform
- **Dream animation**
  Web Animations
- **Revenge of SVG animation**
  Animation Elements



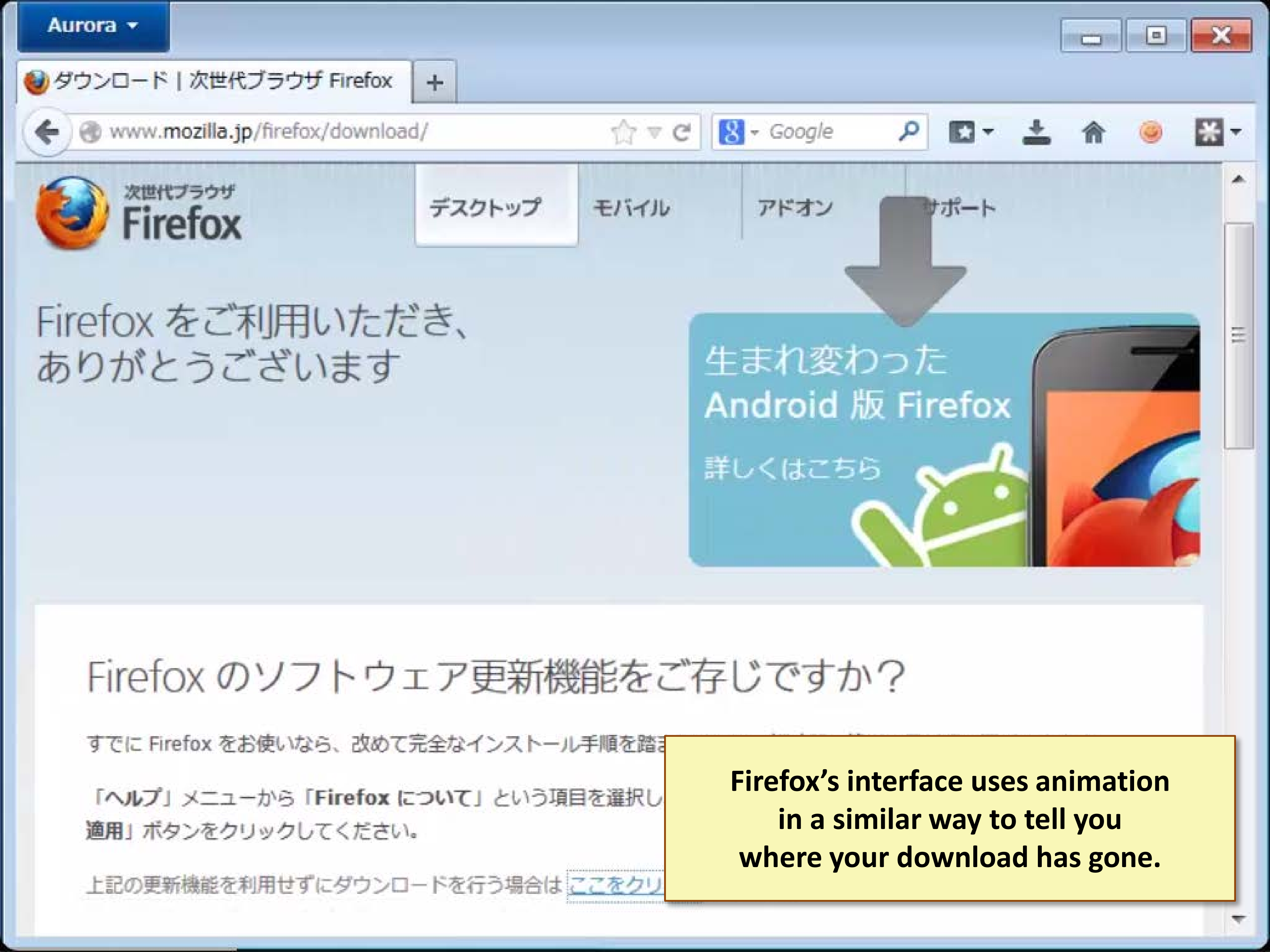
When you think of animation you probably imagine things like this…

Or perhaps this…

But actually there are other ways of using animation!



OSX's genie effect make it easy to understand what has happened to your window and how to get it back



Firefox をご利用いただき、
ありがとうございます

Firefox のソフトウェア更新機能をご存じですか？

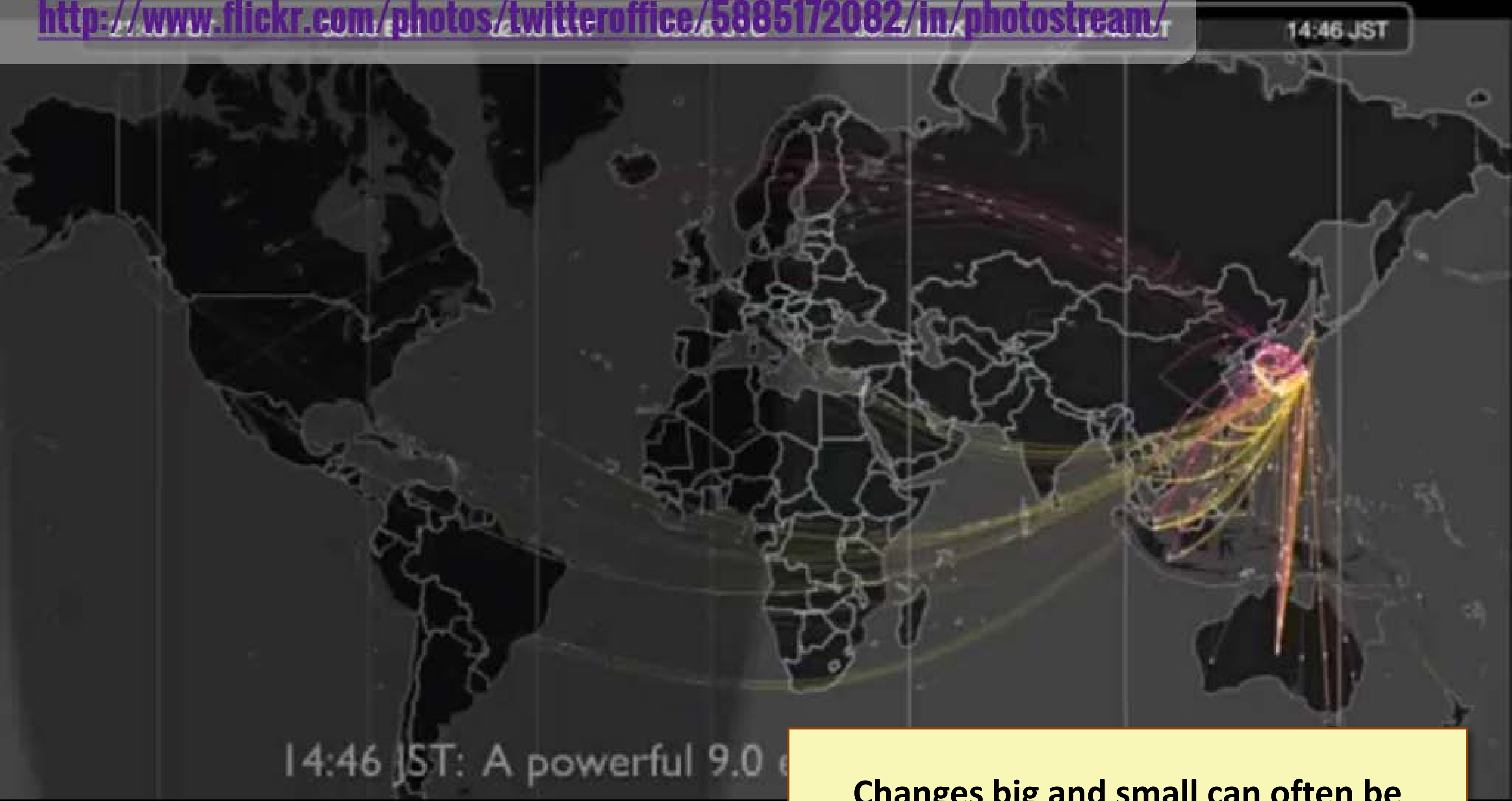
すでに Firefox をお使いなら、改めて完全なインストール手順を踏ま

「ヘルプ」メニューから「**Firefox について**」という項目を選択し
適用」ボタンをクリックしてください。

上記の更新機能を利用せずにダウンロードを行う場合は ここをクリ

**Firefox's interface uses animation
in a similar way to tell you
where your download has gone.**

http://www.flickr.com/photos/twitteroffice/5885172082/in/photostream/



Changes big and small can often be grasped more easily using animation.

Long stories can also be made easy to comprehend using animation.

http://www.youtube.com/watch?v=Yg5BZARVDAs

Animation is essentially using time to convey information.

Part 2

Difficult animation

Problems with the current Web platform

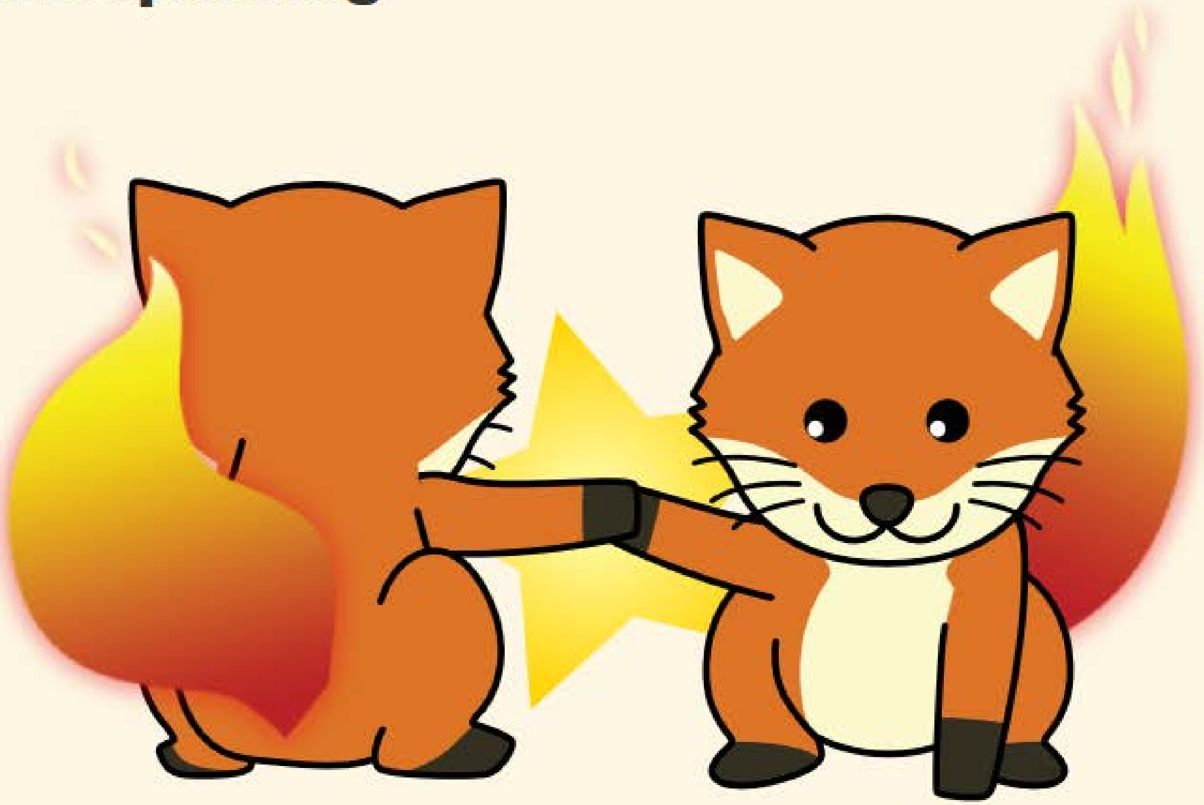
CSS Animations and Transitions

```
h3 {
  animation-name: heading;
  animation-duration: 2s;
  animation-iteration-count: infinite;
}

@keyframes heading {
  50% {
    color: green;
    font-size: 80%;
  }
}
```

CSS Animations and Transitions

- ☹ No sync or sequencing



CSS Animations and Transitions

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence

CSS Animations and Transitions

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence
- ☹ Can't animate same property from two animations

If we apply both these classes to the same element…

`class="spin"`

`class="swell"`

CSS Animations and Transitions

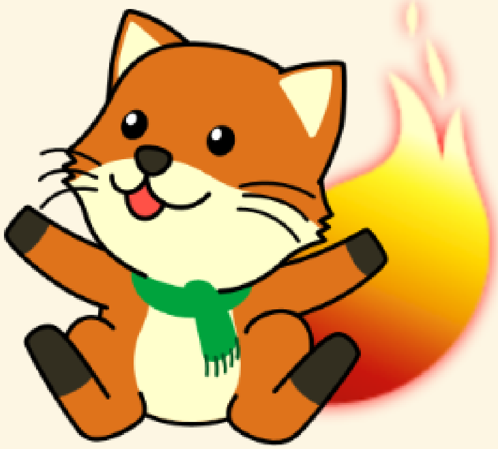
One wins since CSS Animation can’t do addition

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence
- ☹ Can't animate same property from two animations

class=″spin″

class=″swell″

class=″spin swell″



CSS Animations and Transitions

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence
- ☹ Can't animate same property from two animations

If we make up a single animation combining both effects the timing won’t match and we have to prepare all possible combinations in advance

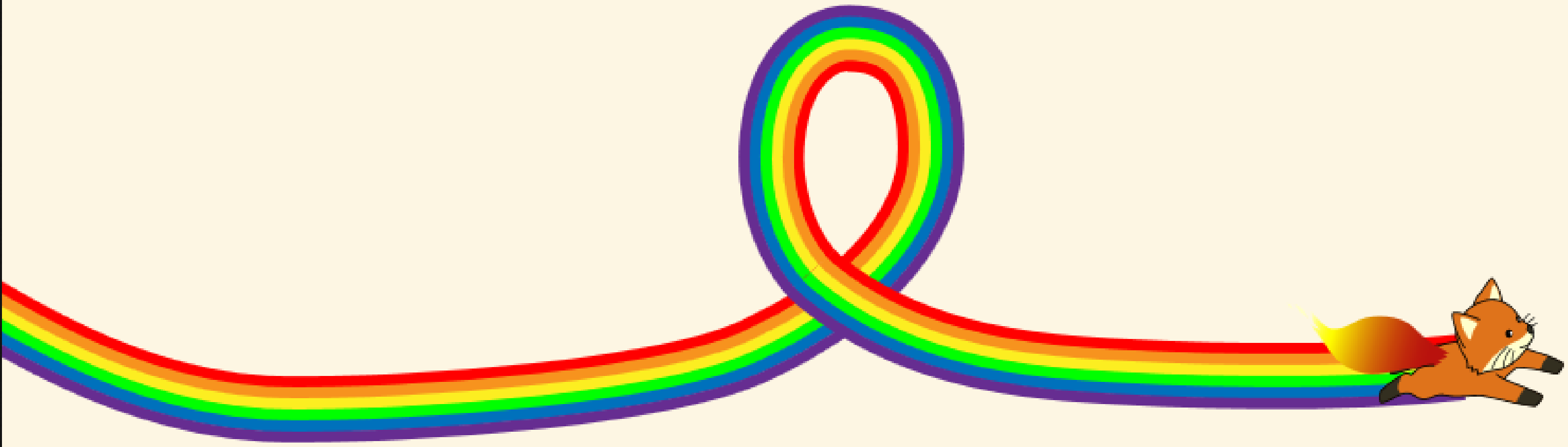
`class="spin"`

`class="swell"`

`class="spin-swell"`

CSS Animations and Transitions

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence
- ☹ Can't animate same property from two animations
- ☹ No motion along a path



CSS Animations and Transitions

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence
- ☹ Can't animate same property from two animations
- ☹ No motion along a path
- ☹ Can only animate CSS properties
    - ✗ path, scrollbar, canvas, WebGL, <video>

CSS Animations and Transitions

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence
- ☹ Can't animate same property from two animations
- ☹ No motion along a path
- ☹ Can only animate CSS properties
  - ╳ path, scrollbar, canvas, WebGL, <video>

We can’t animate Foxkeh’s eyes like this

CSS Animations and Transitions

- ☹ No sync or sequencing
- ☹ No seek, pause, reverse on a sequence
- ☹ Can't animate same property from two animations
- ☹ No motion along a path
- ☹ Can only animate CSS properties
    - ✗ path, scrollbar, canvas, WebGL, `<video>`
- ☹ Can't use Javascript to debug etc.

SVG Animation

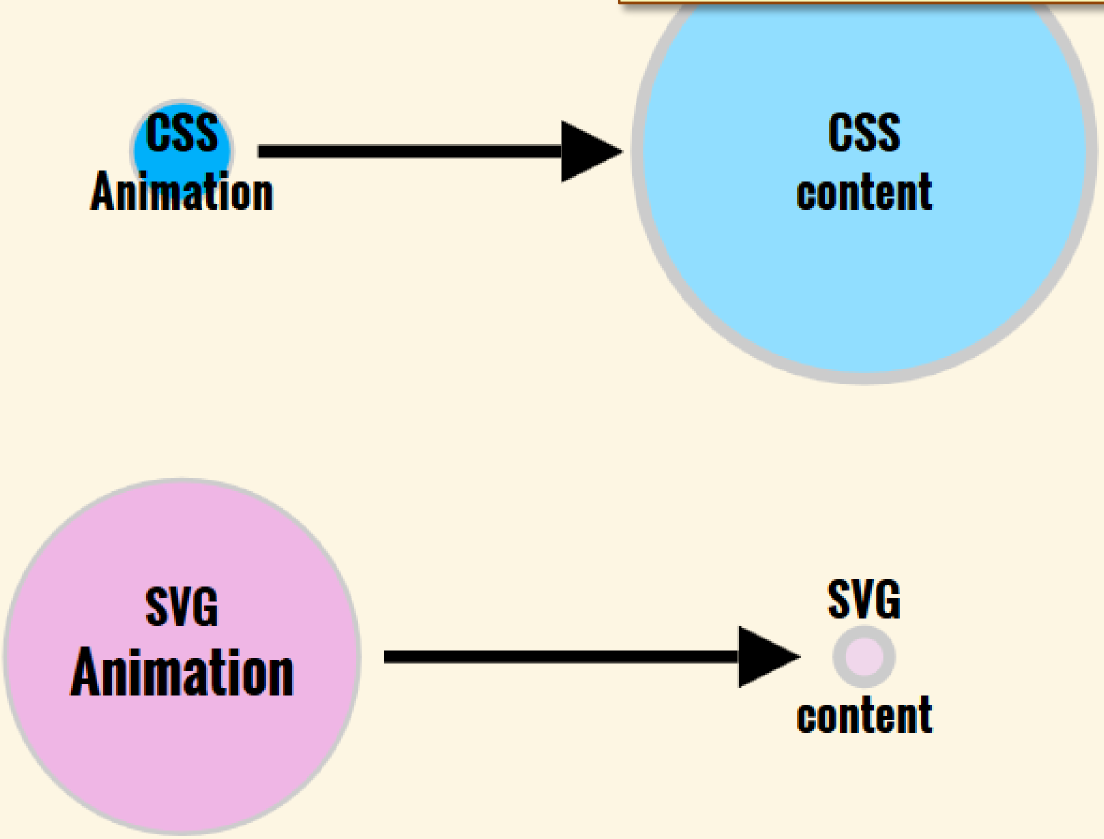
SVG Animation actually has more features…

CSS Animation

SVG Animation

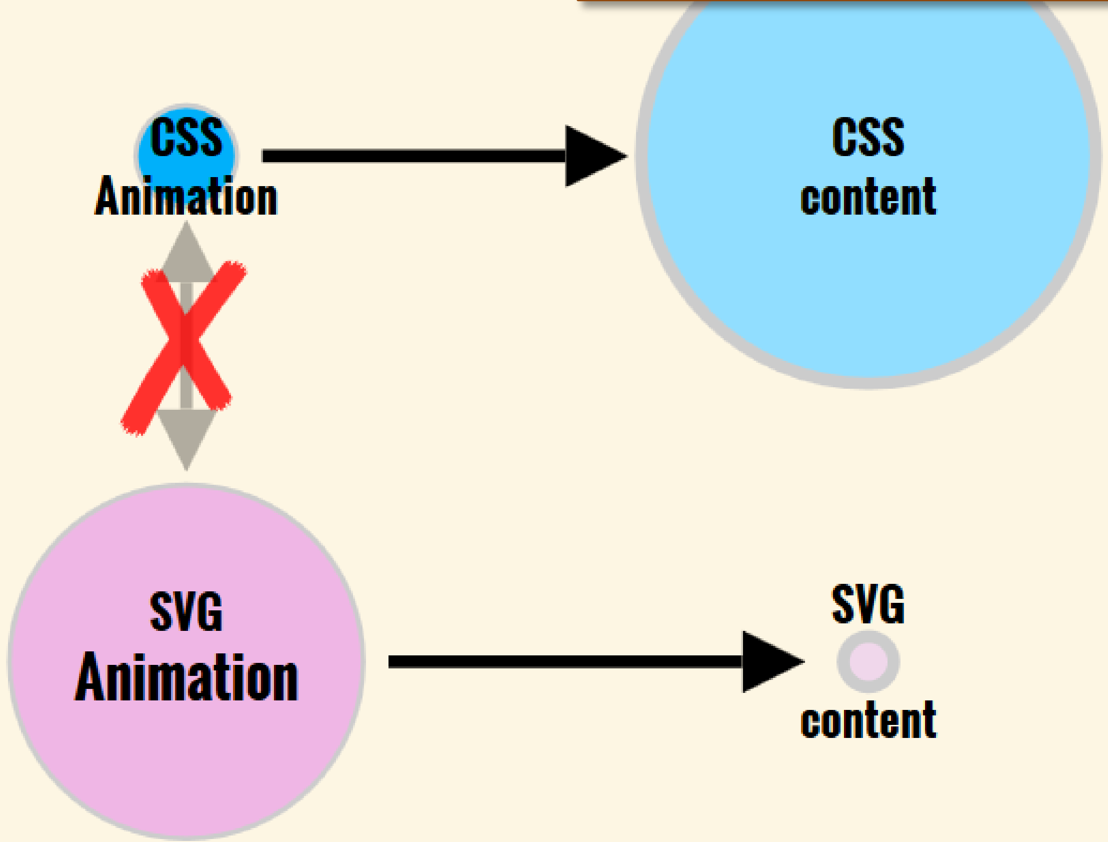
SVG Animation

But CSS has much more content.



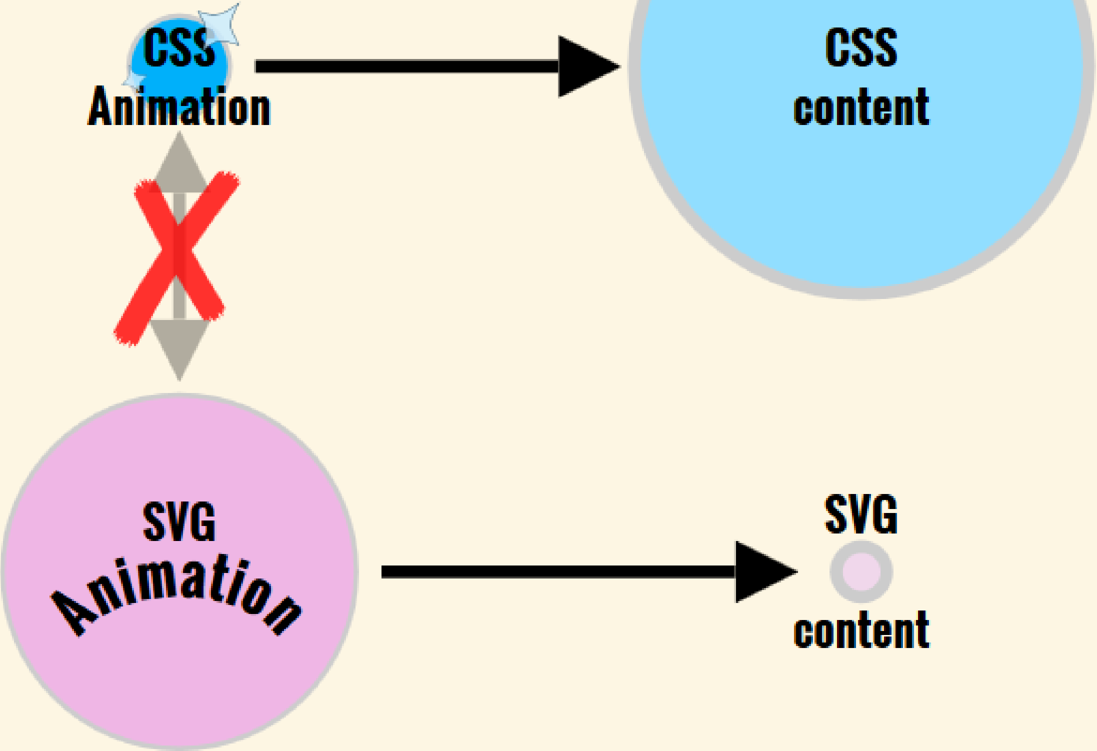
SVG Animation

In most browsers the implementation of CSS Animation and SVG Animation is completely separate.



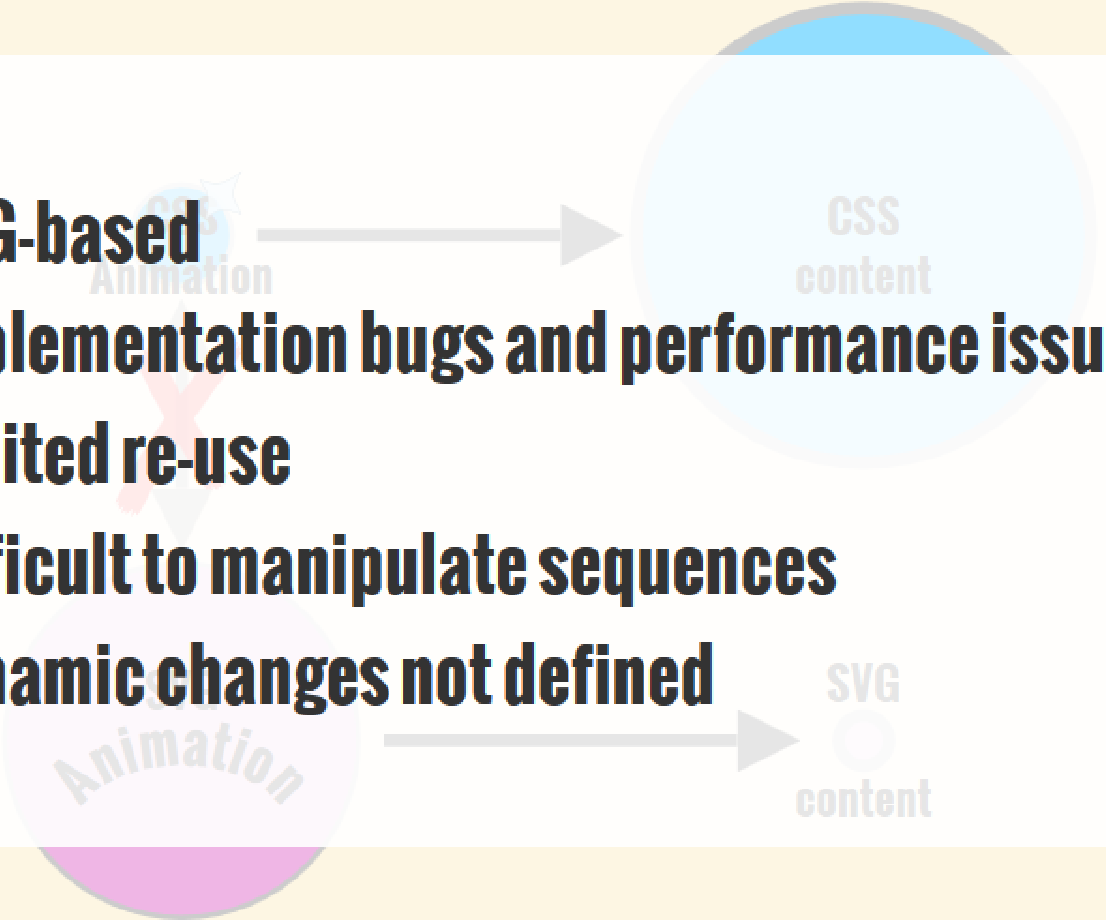
SVG Animation

As a result CSS is more optimised and correctly and widely implemented

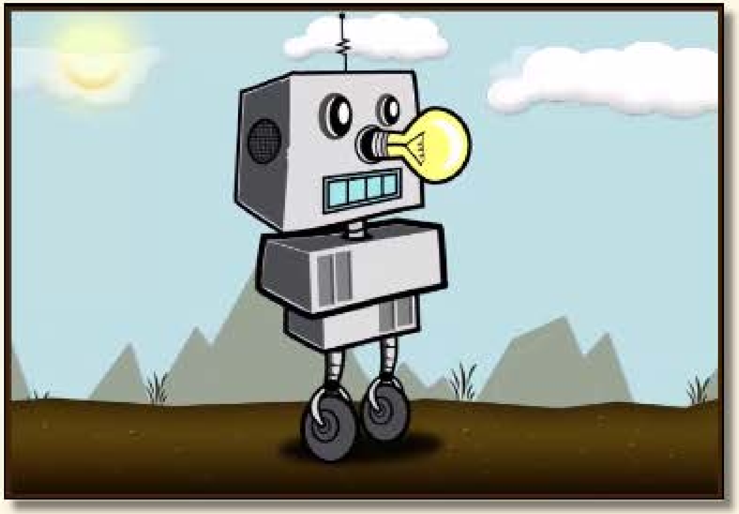


SVG Animation

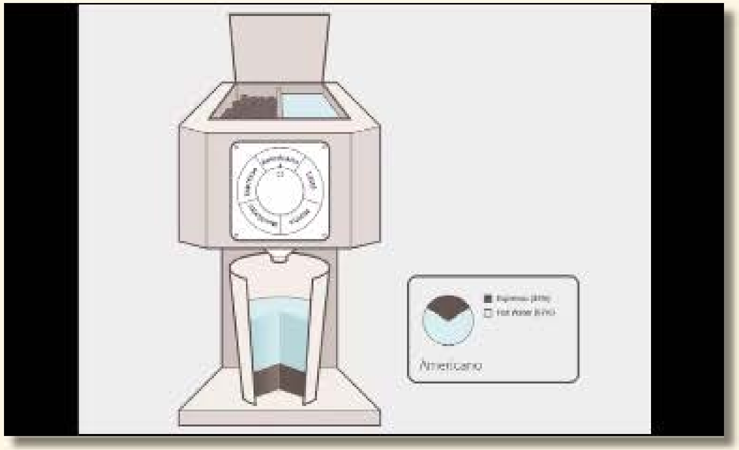
- ☹ SVG-based
- ☹ Implementation bugs and performance issues
- ☹ Limited re-use
- ☹ Difficult to manipulate sequences
- ☹ Dynamic changes not defined



Javascript



jQuery + HTML http://robot.anthonycalzadilla.com/



Snap.svg + SVG http://snapsvg.io/demos/#coffee

Canvas http://www.cuttherope.ie/

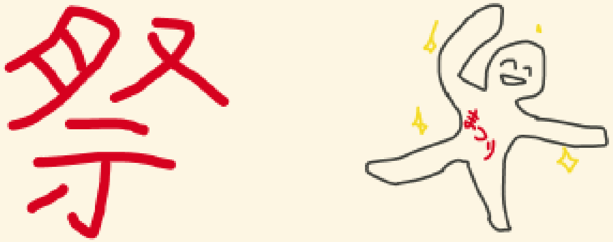
WebGL http://www.unrealengine.com/html5/

Javascript

- ☹ Runs on main thread

(in so far as making changes to CSS properties is concerned)

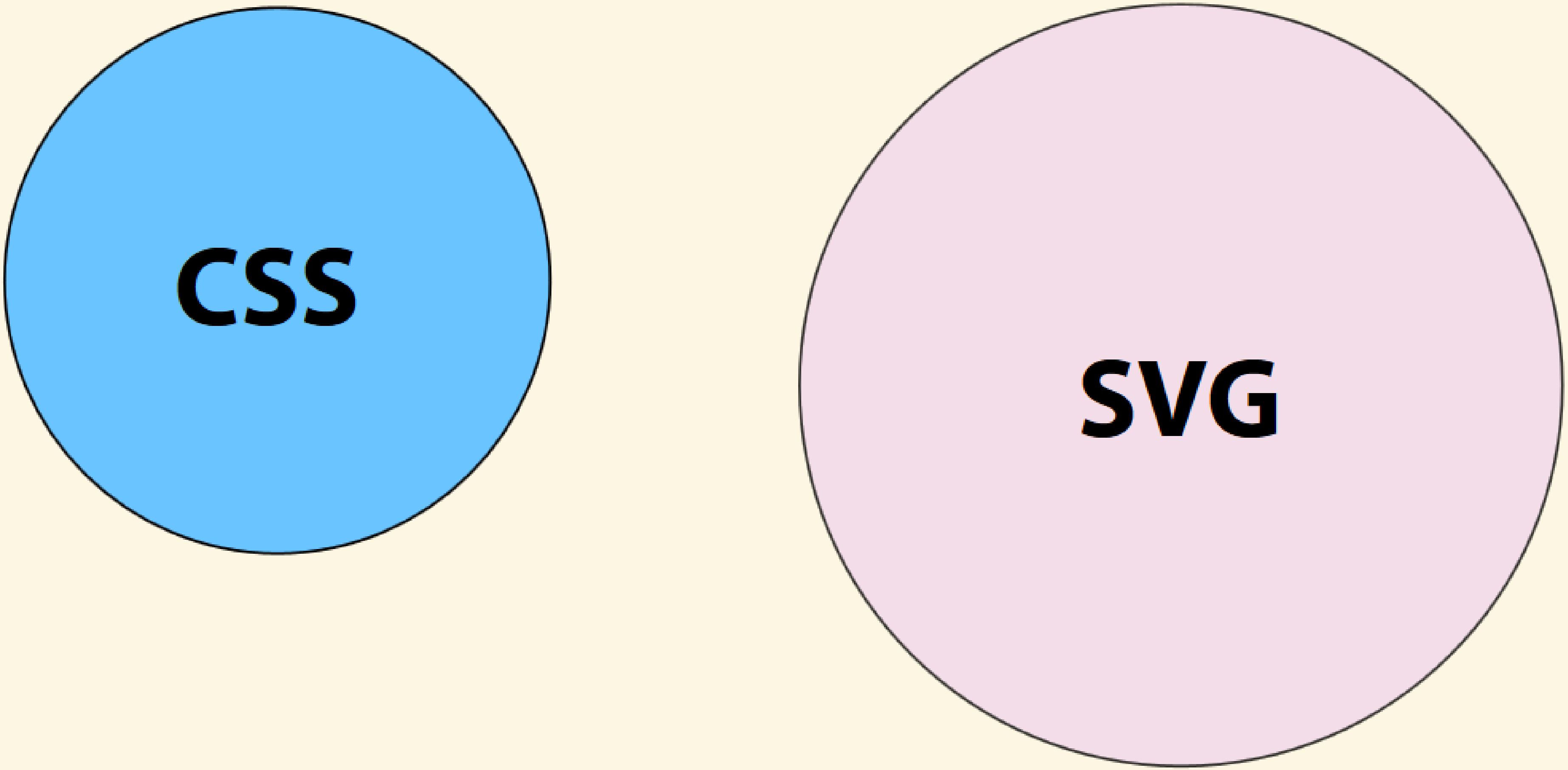
- ☹ Can't add to SVG animation
- ☹ Places where Javascript can't be used
  - ○ <img src="anim.svg">, background-image: url("anim.svg")など
  - ○ SVG-in-OpenType (emoji etc.)
  - ○ <iframe sandbox>



Part 3

Animation of your dreams

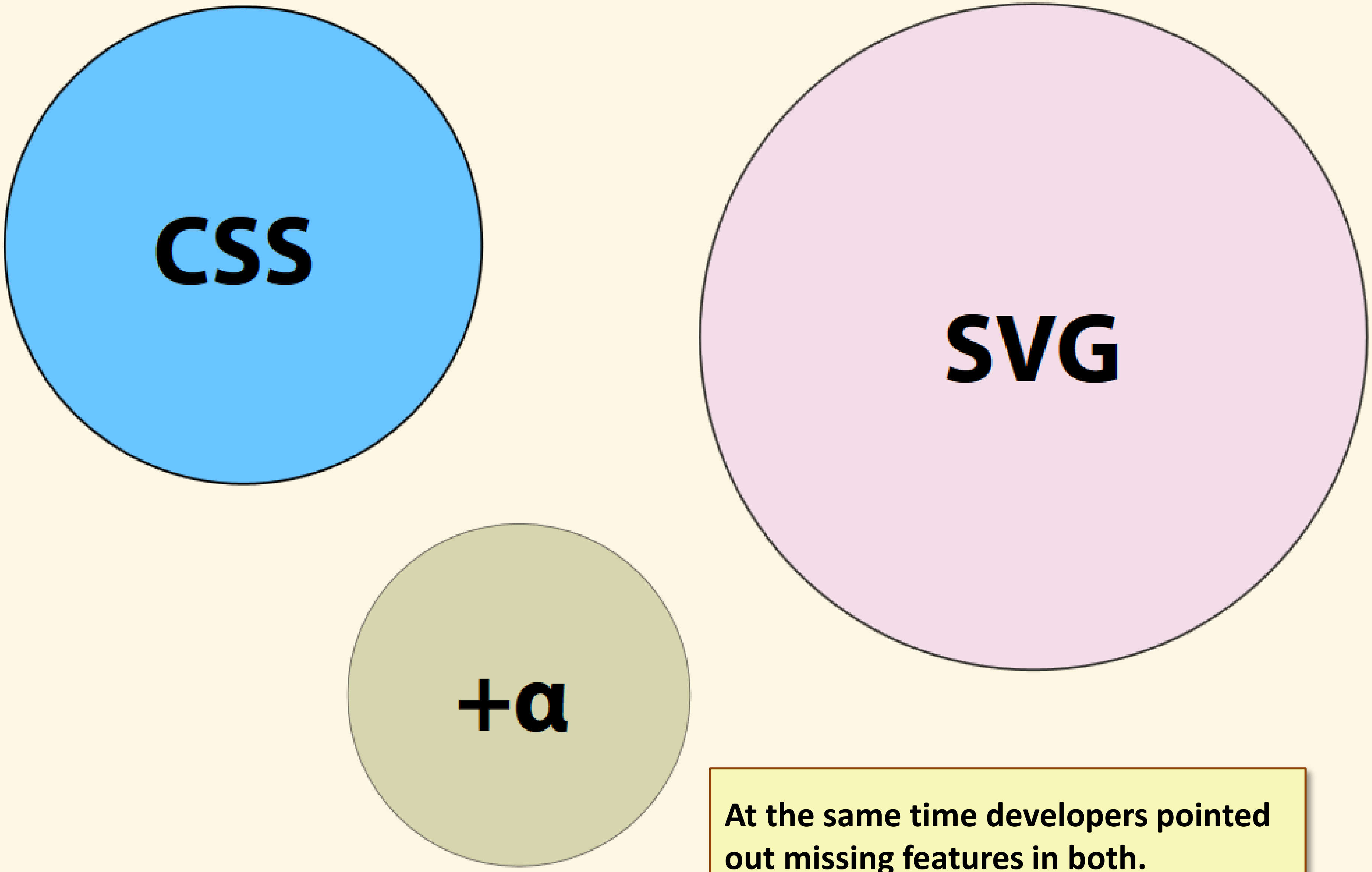
Web Animations



CSS

SVG

Some people suggested we shouldn’t have two animation models for the Web.

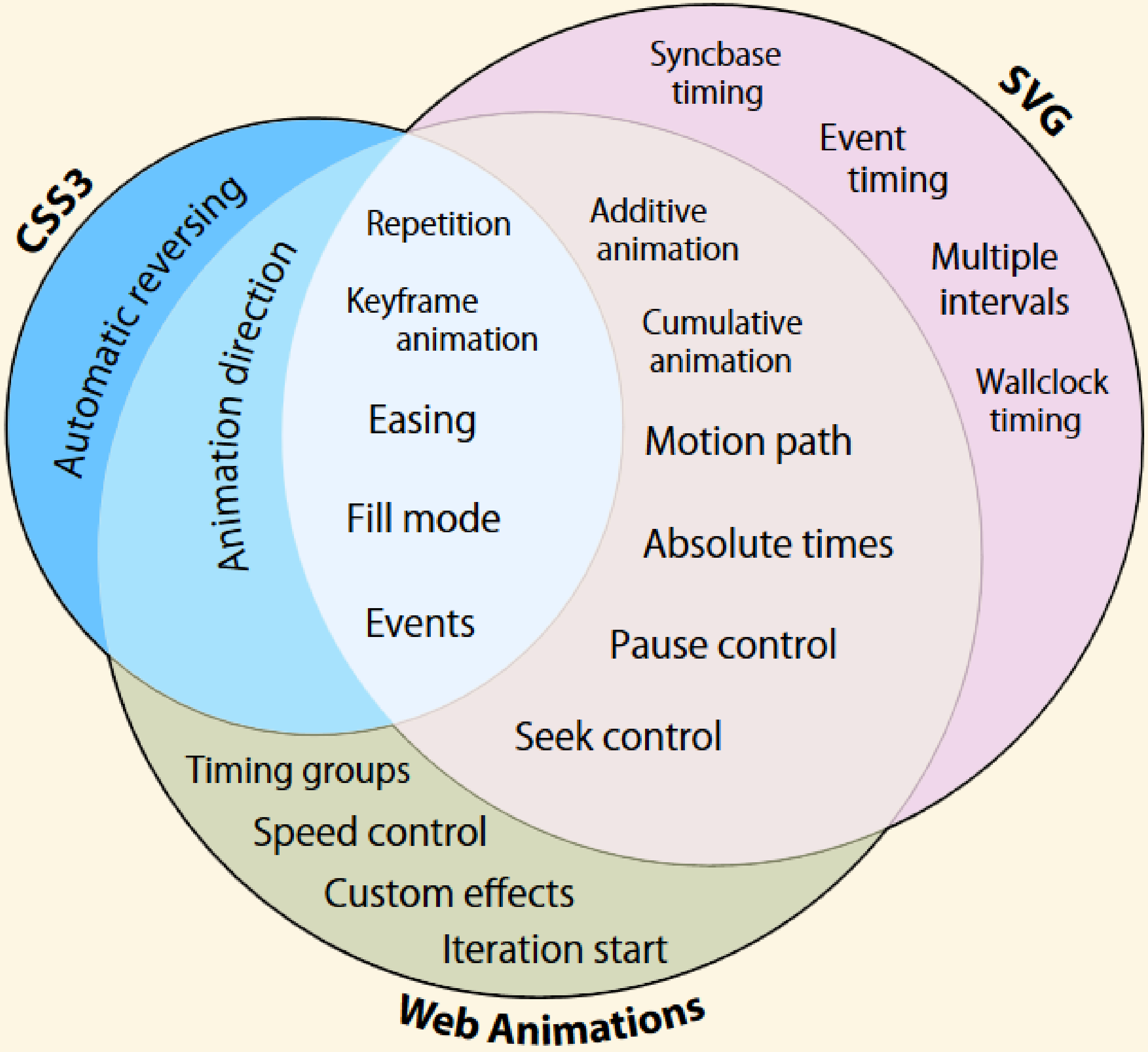


CSS

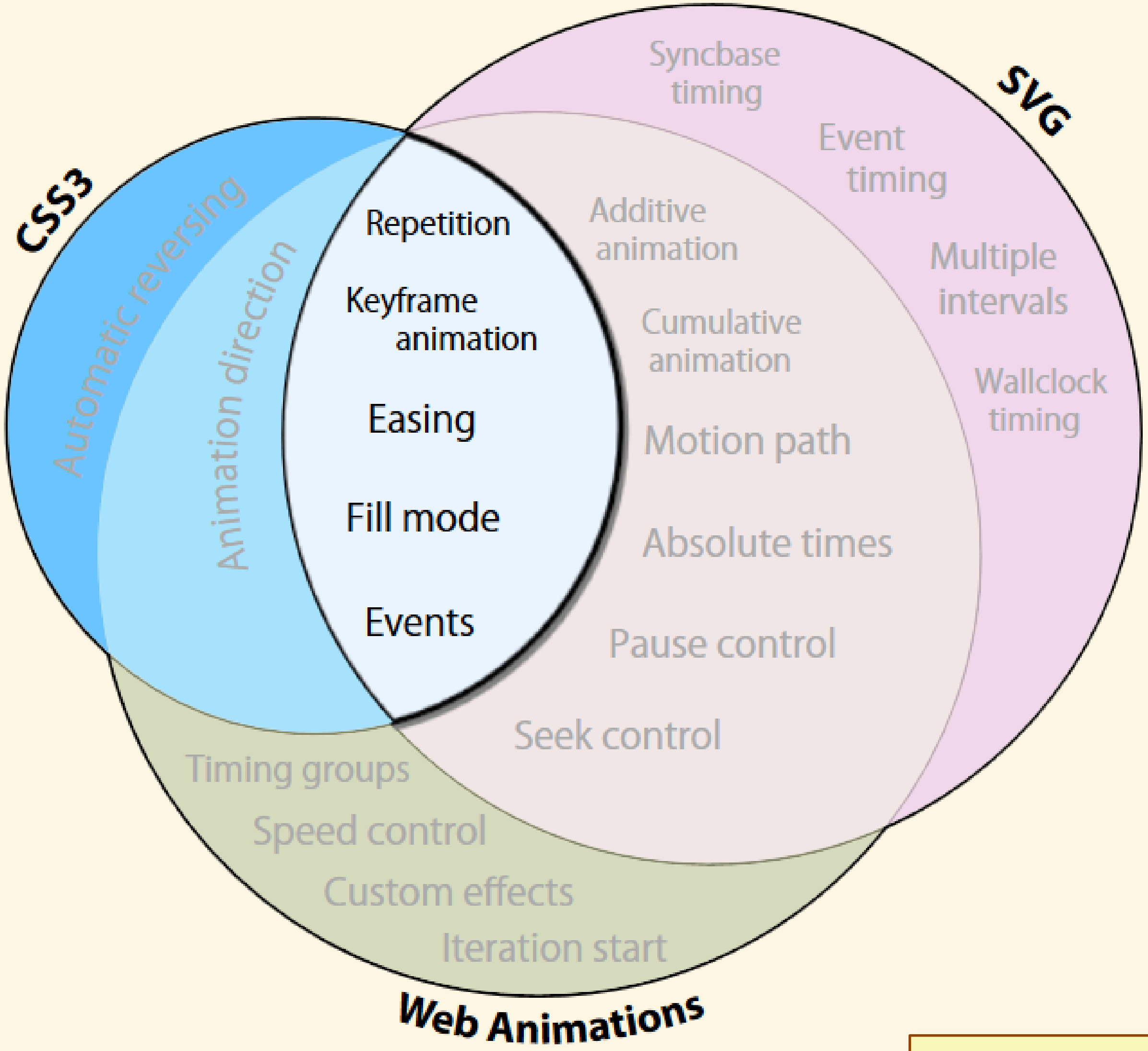
SVG

+α

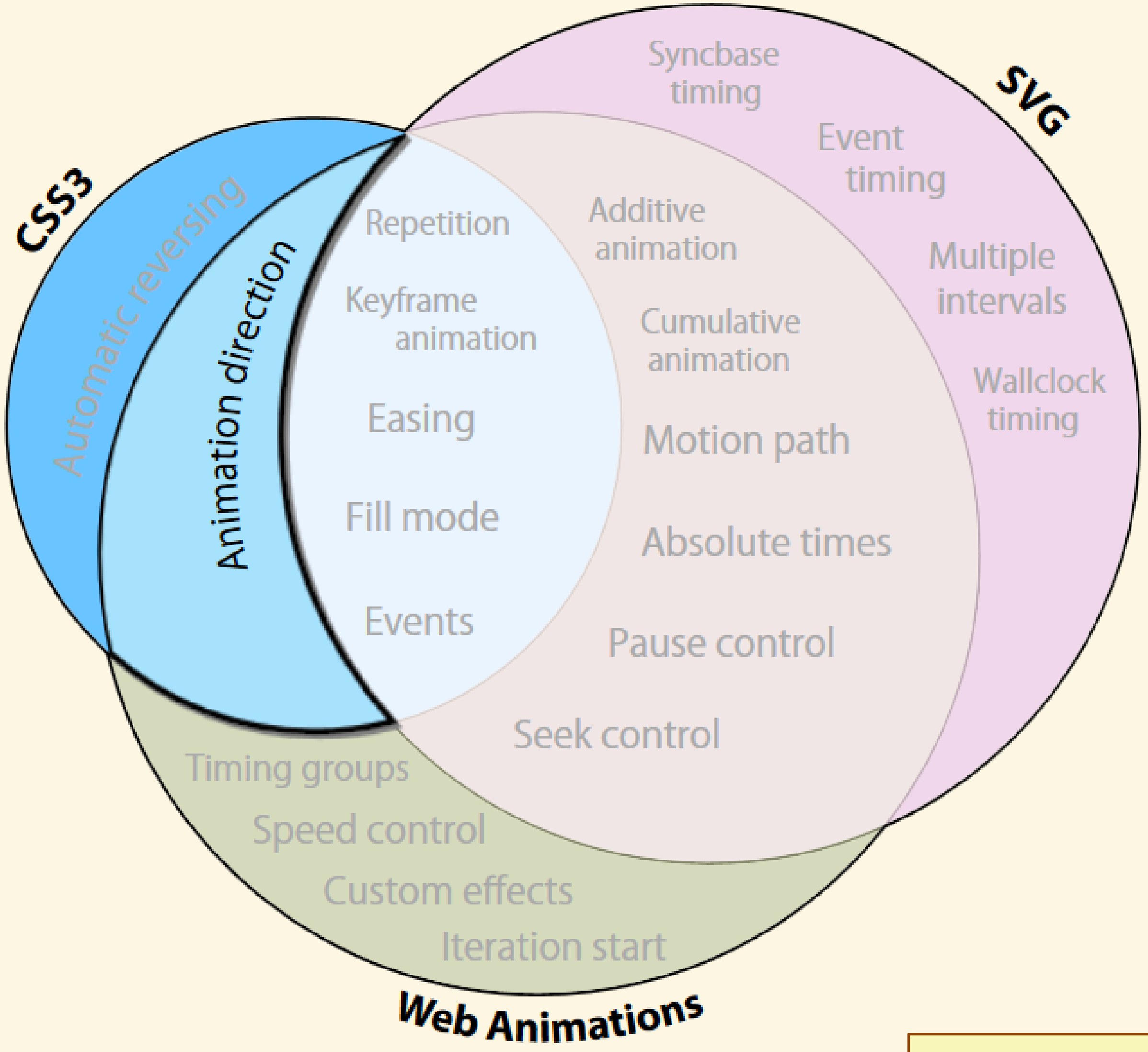
At the same time developers pointed out missing features in both.



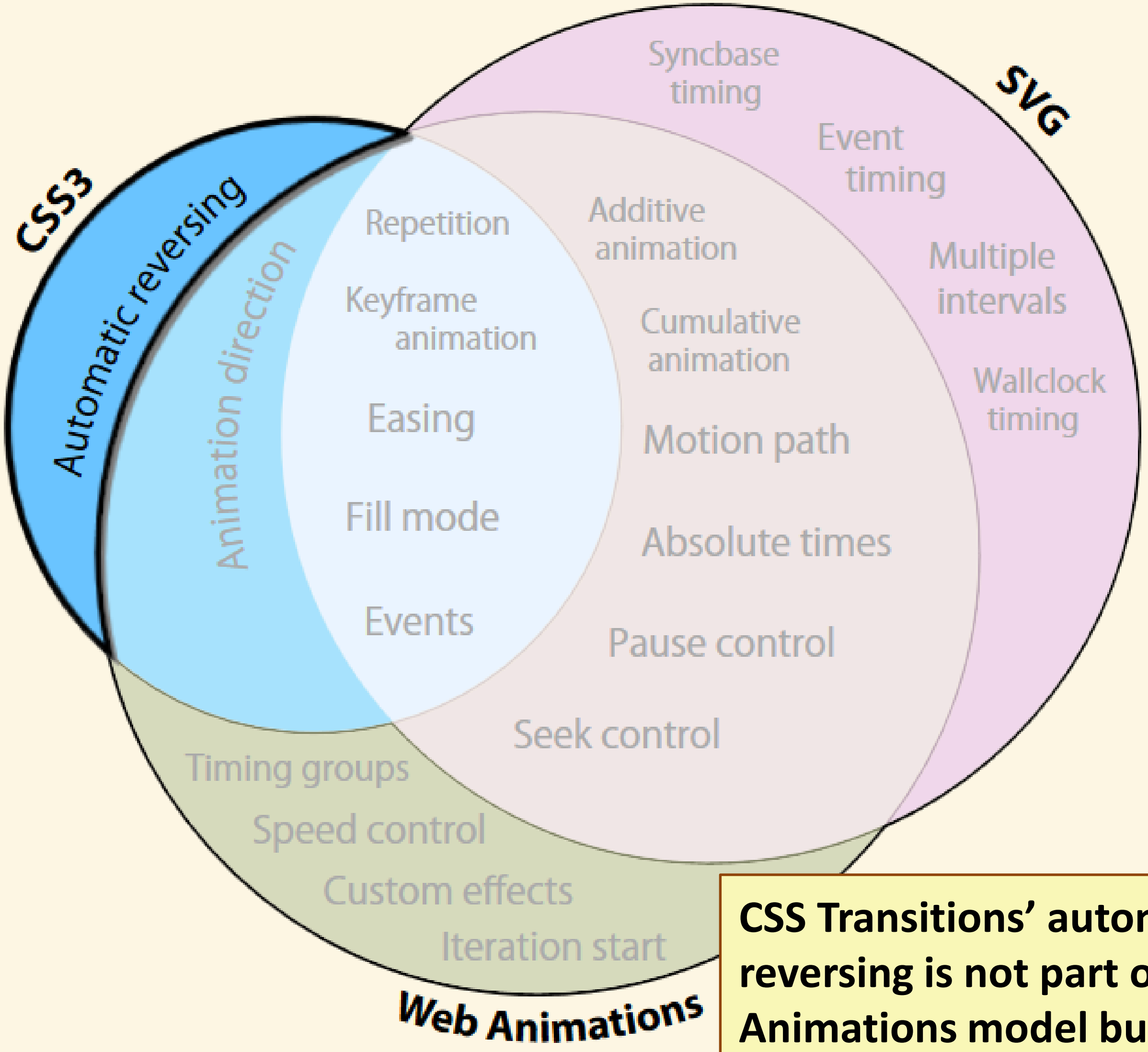
So we did this.



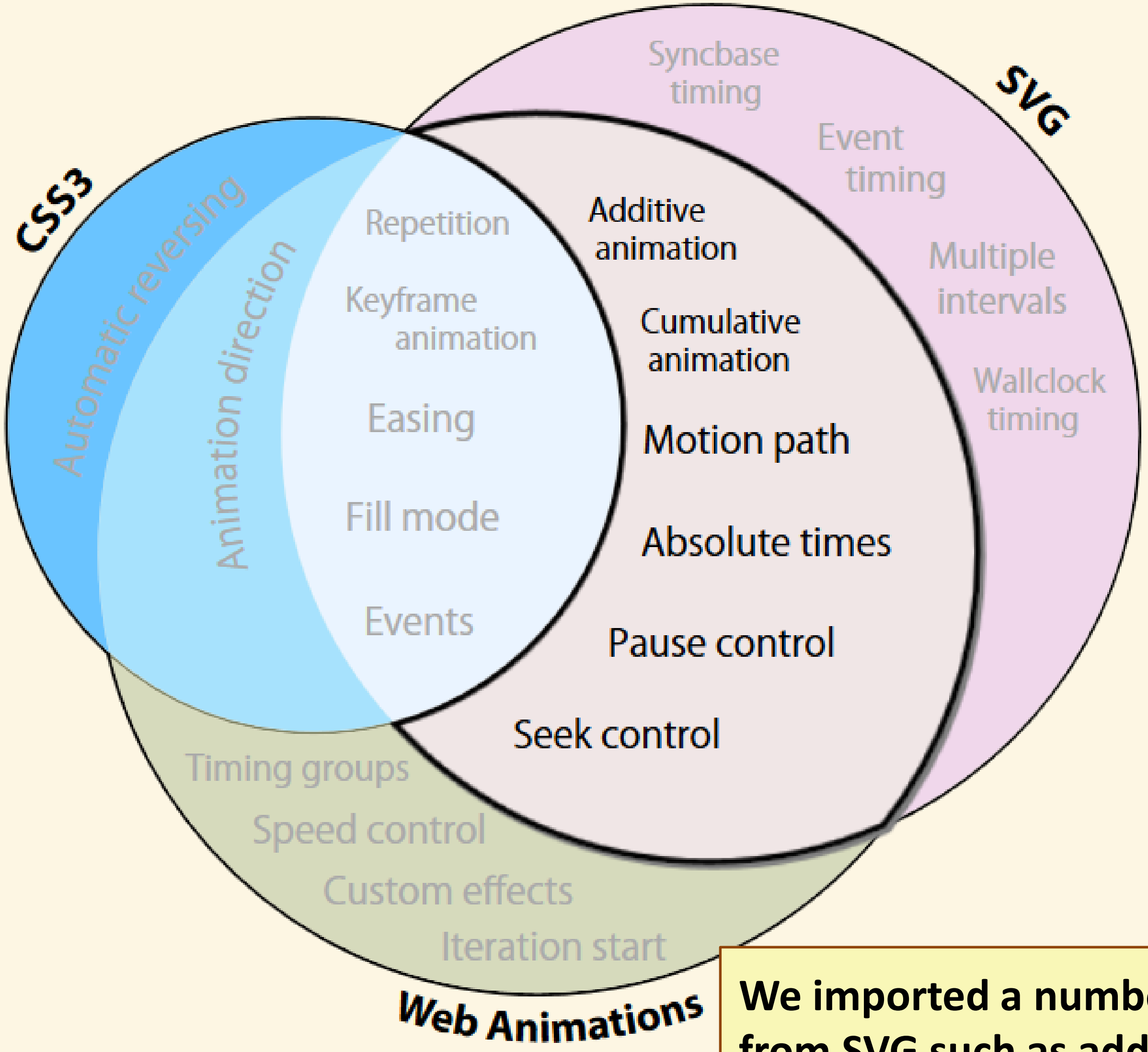
Features we already had in both CSS and SVG.



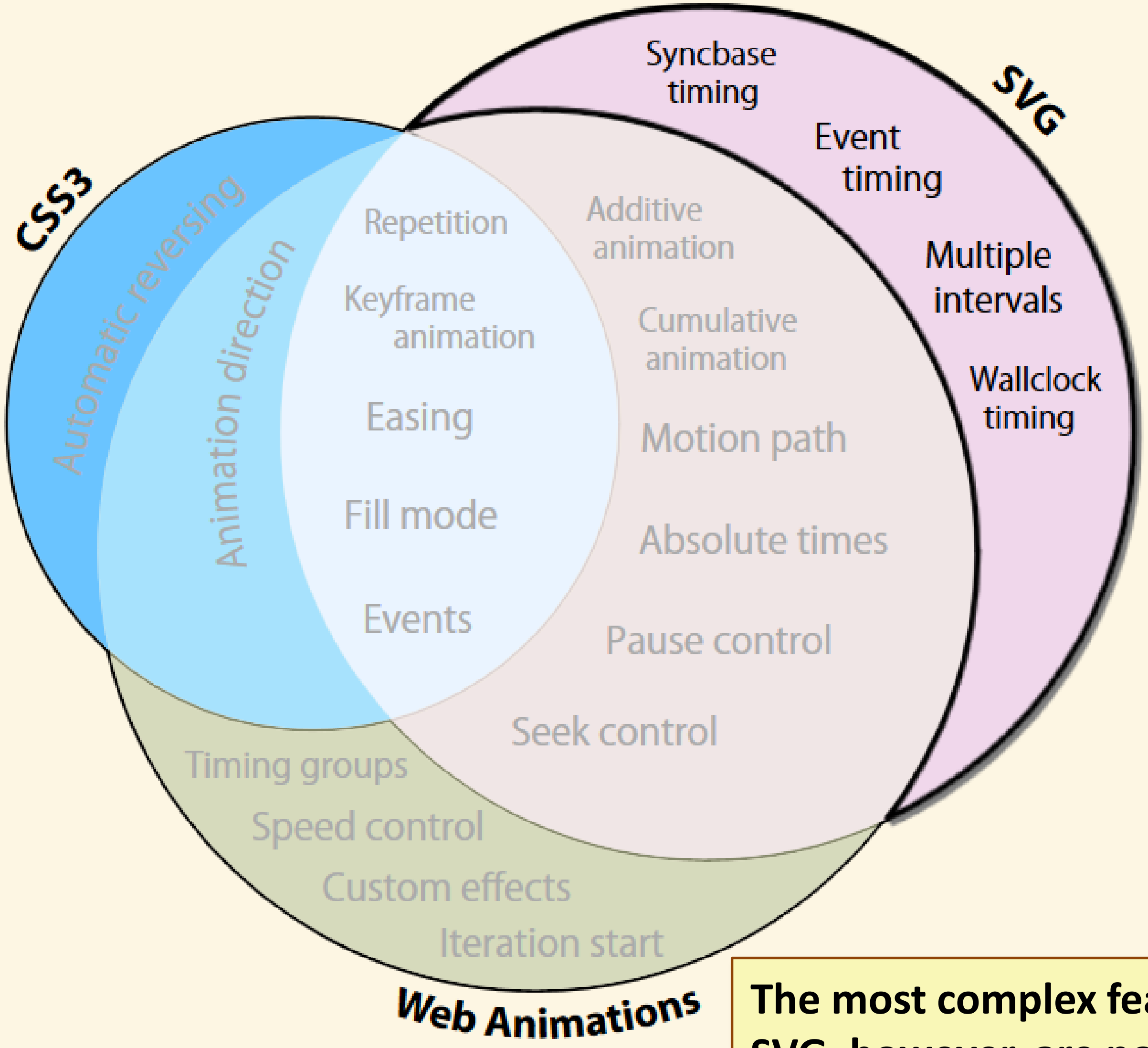
From CSS we imported animation direction.



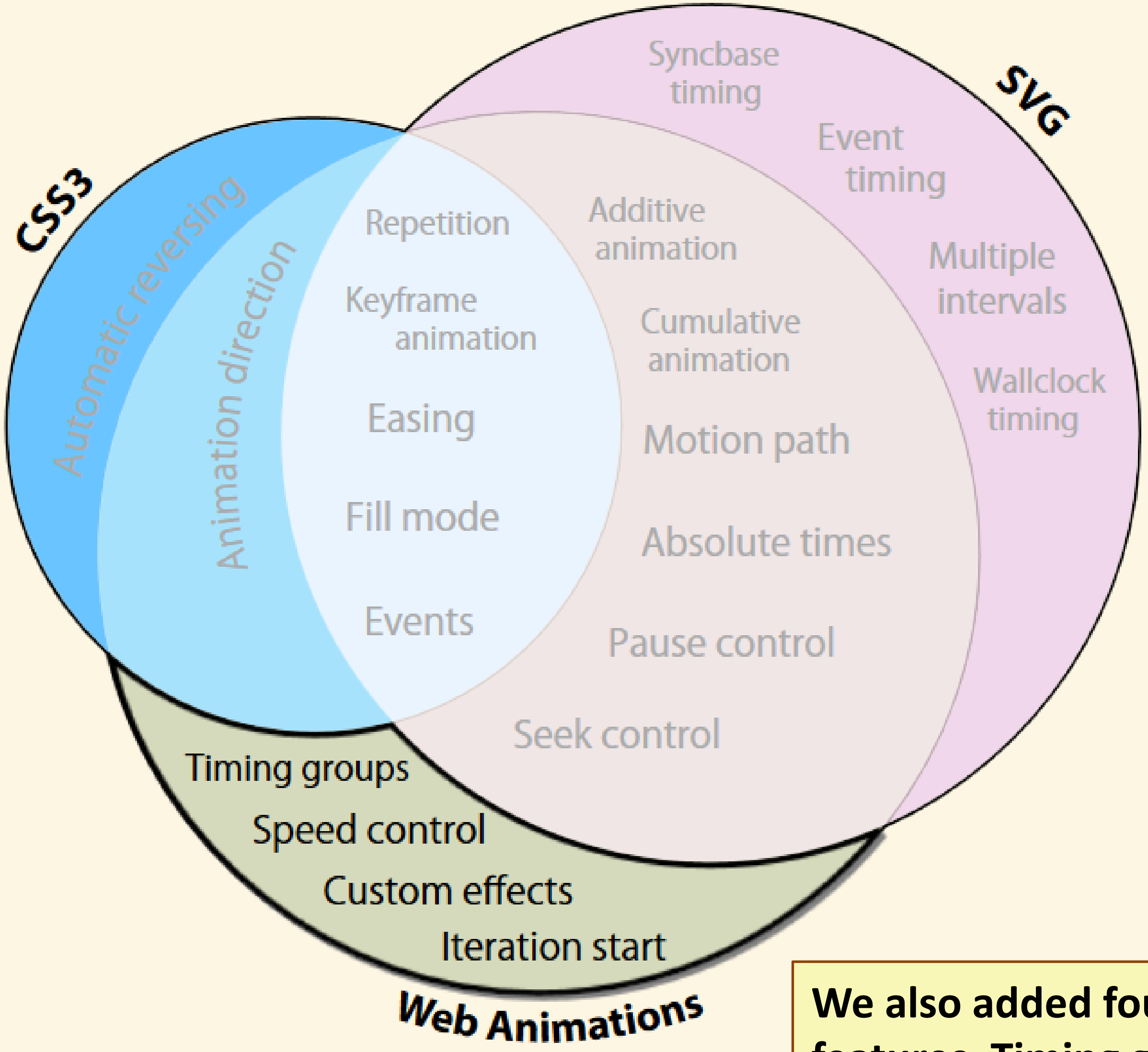
CSS Transitions' automatic reversing is not part of the Web Animations model but can be realized using other features of the model.



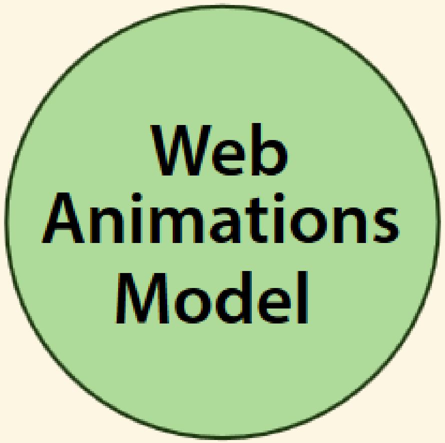
We imported a number of features from SVG such as additive animation, motion path animation and seek control.



The most complex features from SVG, however, are not part of the model but are included in the SVG bindings (“Animation Elements”).

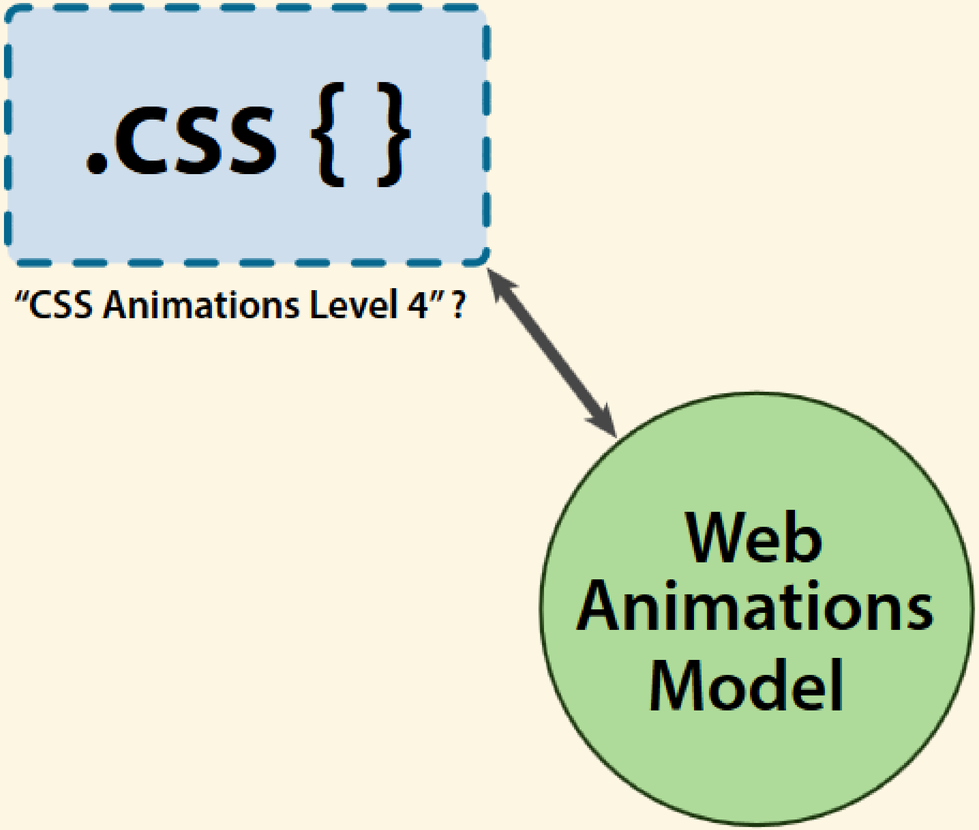


We also added four new features. Timing groups and custom effects are particularly powerful.

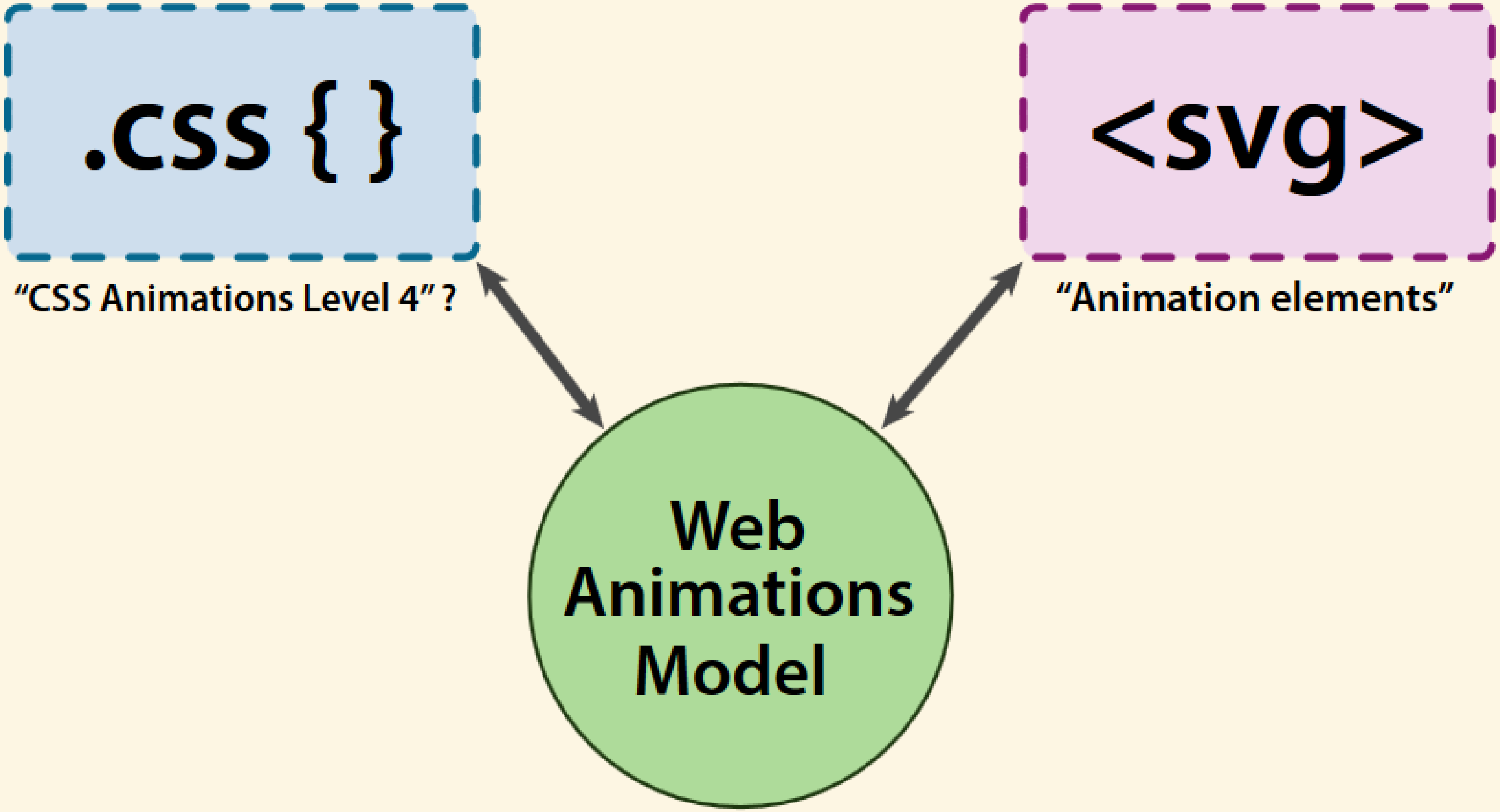


Web Animations Model

Web Animations is fundamentally an abstract model.



CSS syntax and mapping to the model will be defined in a separate spec.



.css { }

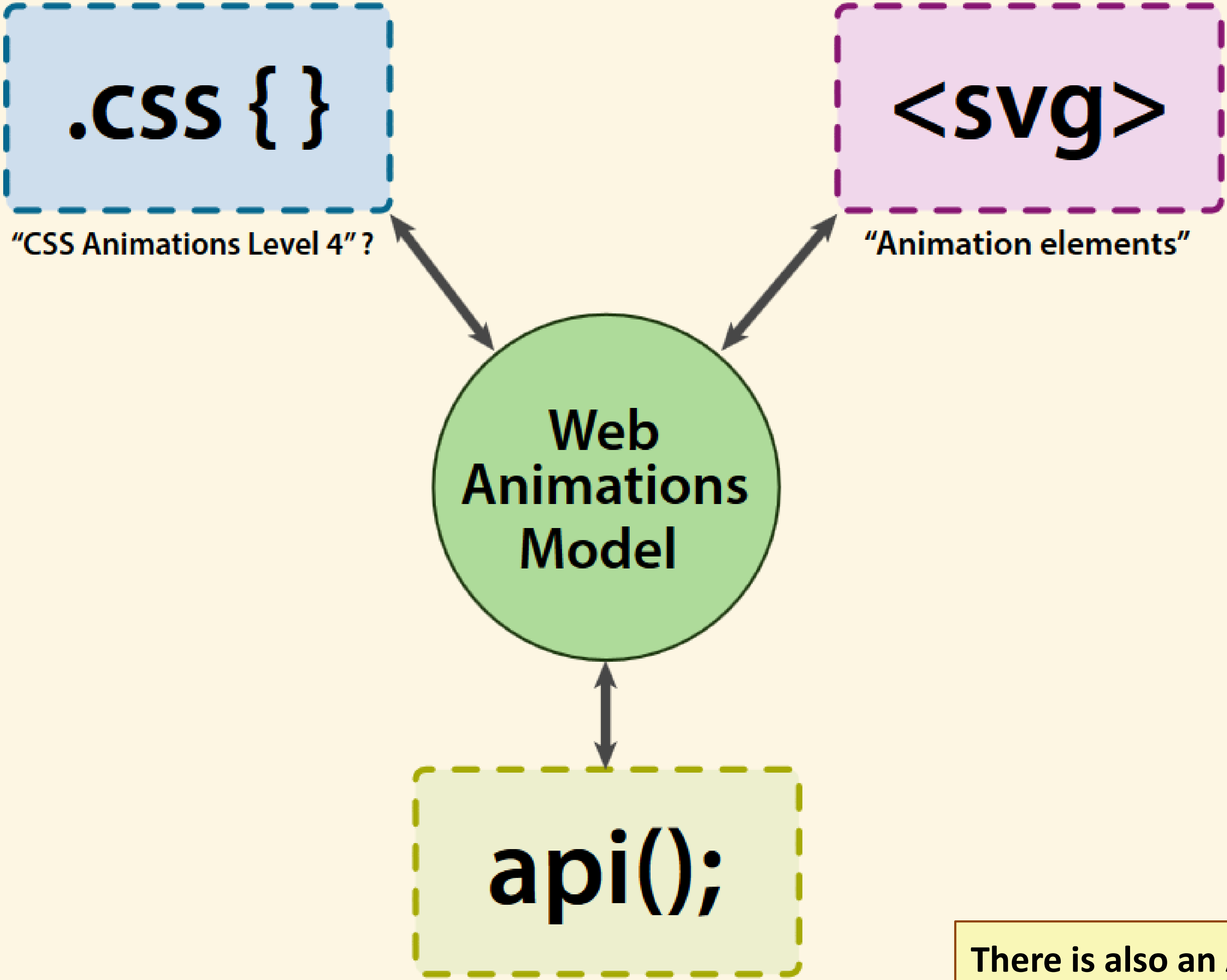
“CSS Animations Level 4” ?

<svg>

“Animation elements”

Web Animations Model

Likewise, SVG syntax and mapping will be defined in the Animation Elements spec.



.css { }

“CSS Animations Level 4” ?

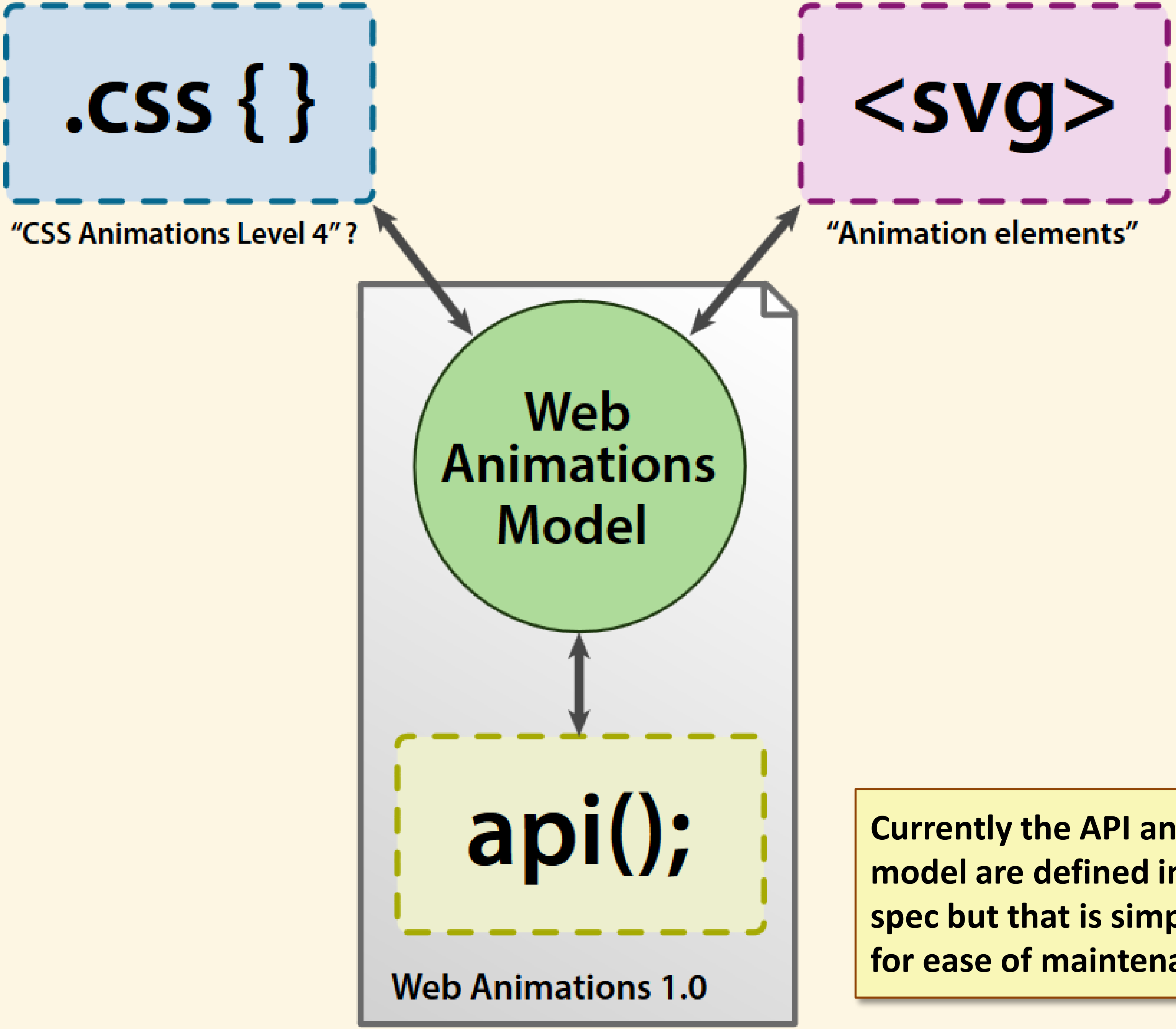
<svg>

“Animation elements”

Web Animations Model

api();

There is also an API onto the model.



.css { }

“CSS Animations Level 4” ?

<svg>

“Animation elements”

Web Animations Model

api();

Web Animations 1.0

Currently the API and model are defined in one spec but that is simply for ease of maintenance.

Basic interpolation

```
Element.animate(<properties>, <time>)
```

Basic interpolation

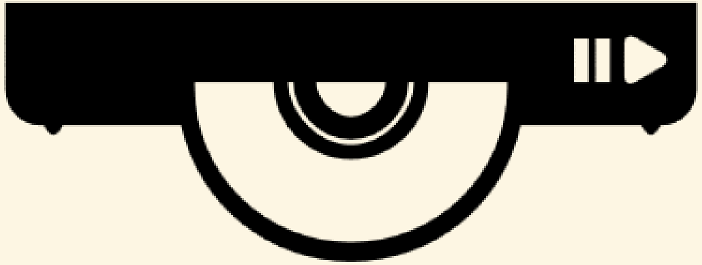
```
Element.animate(
  { transform: 'rotate(360deg)' }, 1)
```

```
Element.animate(
  { transform: 'rotate(360deg)' },
  { duration: 1,
    easing: 'ease-in-out',
    direction: 'alternate',
    iterations: Infinity }
```

Players

- cancel()
- finish()
- play()
- pause()
- reverse()
- currentTime
- playbackRate
- paused

http://brian.sol1.net/svg/2013/07/25/players-wanted-the-pause-and-seek-game/

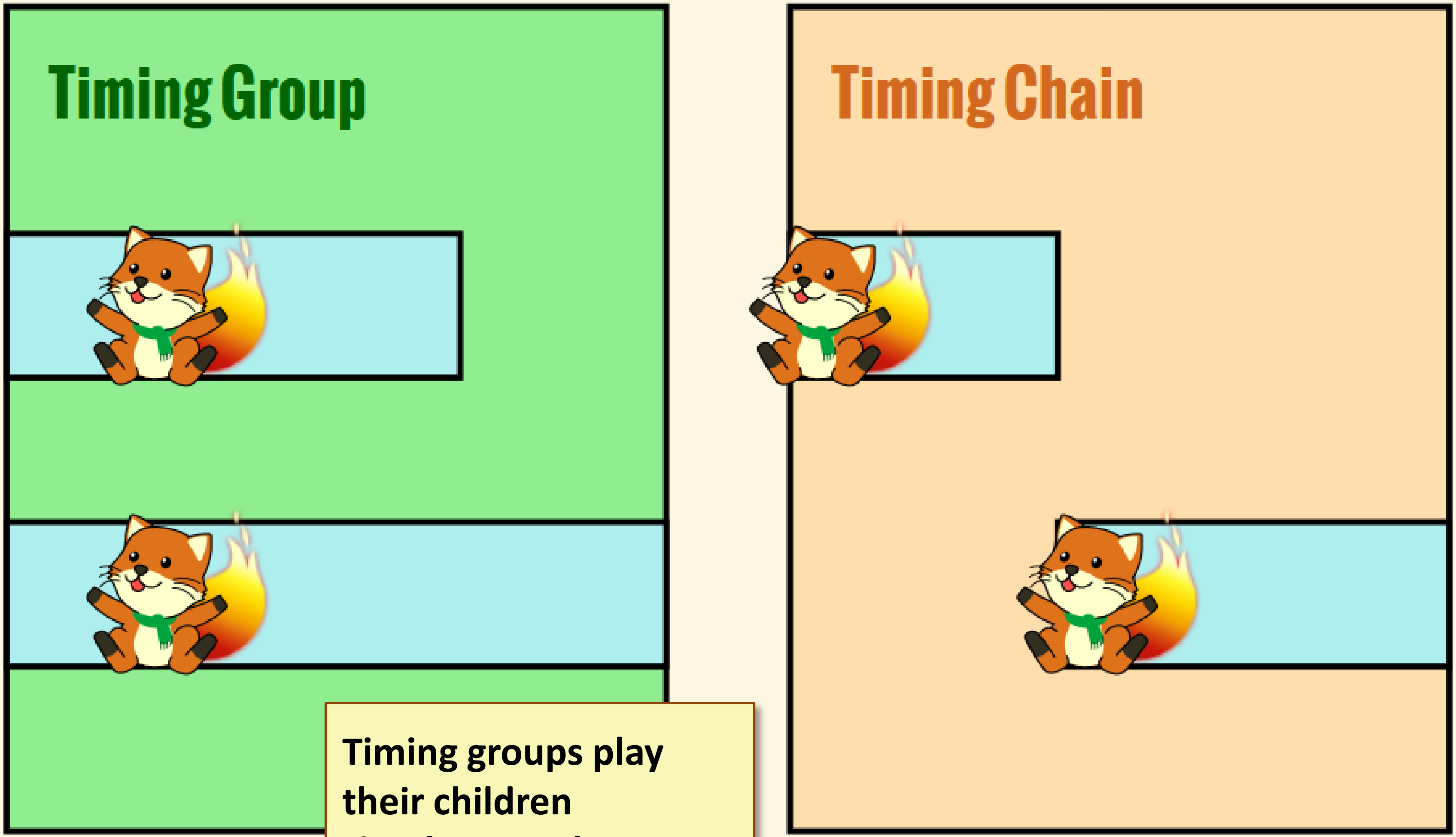


DVD Player designed by Dan Hetteix from The Noun Project

Players are used to control playback of animations and timing groups.

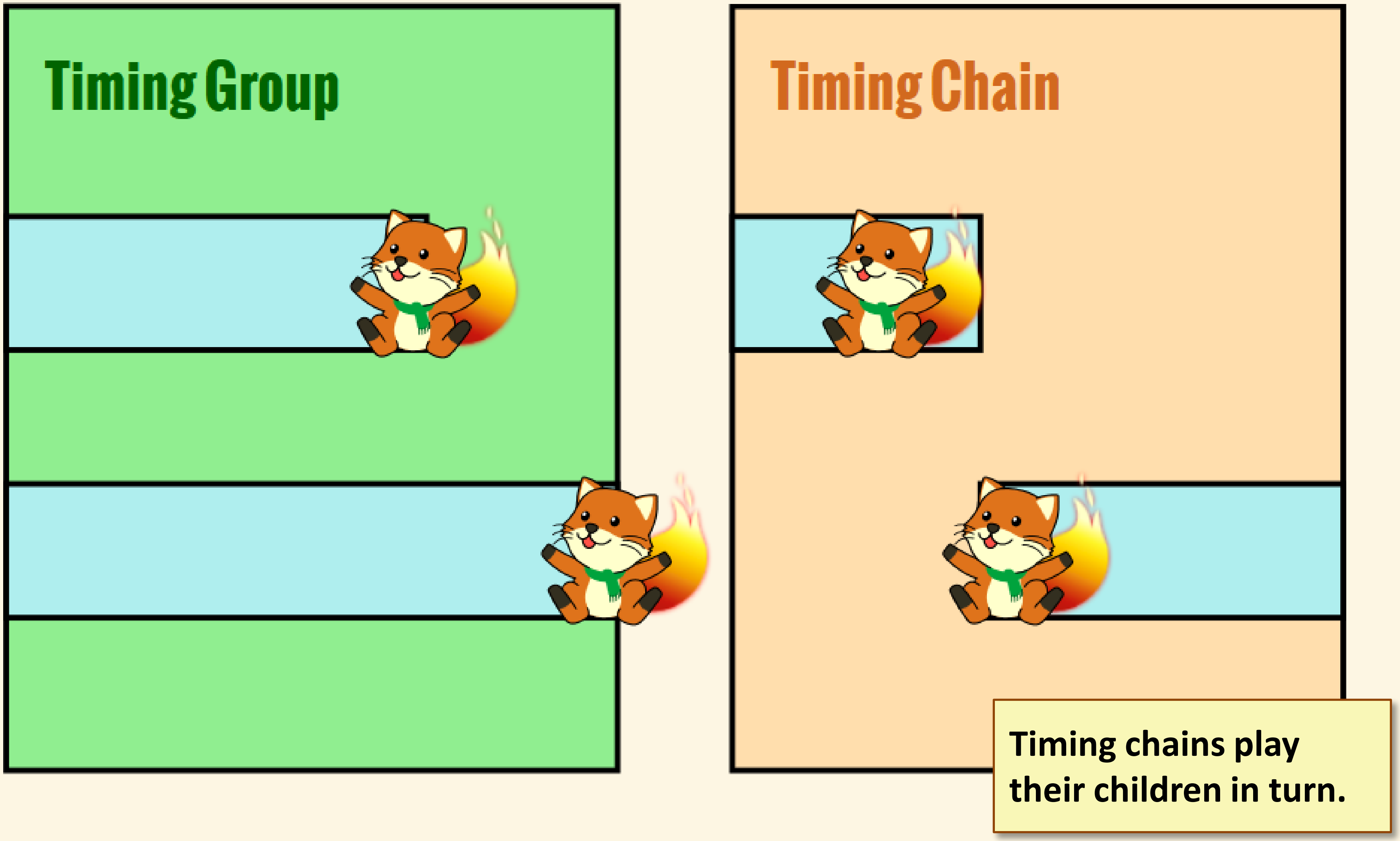
Timing groups

There are two types of timing groups: timing groups and timing chains (although these names may change).



Timing groups play their children simultaneously.

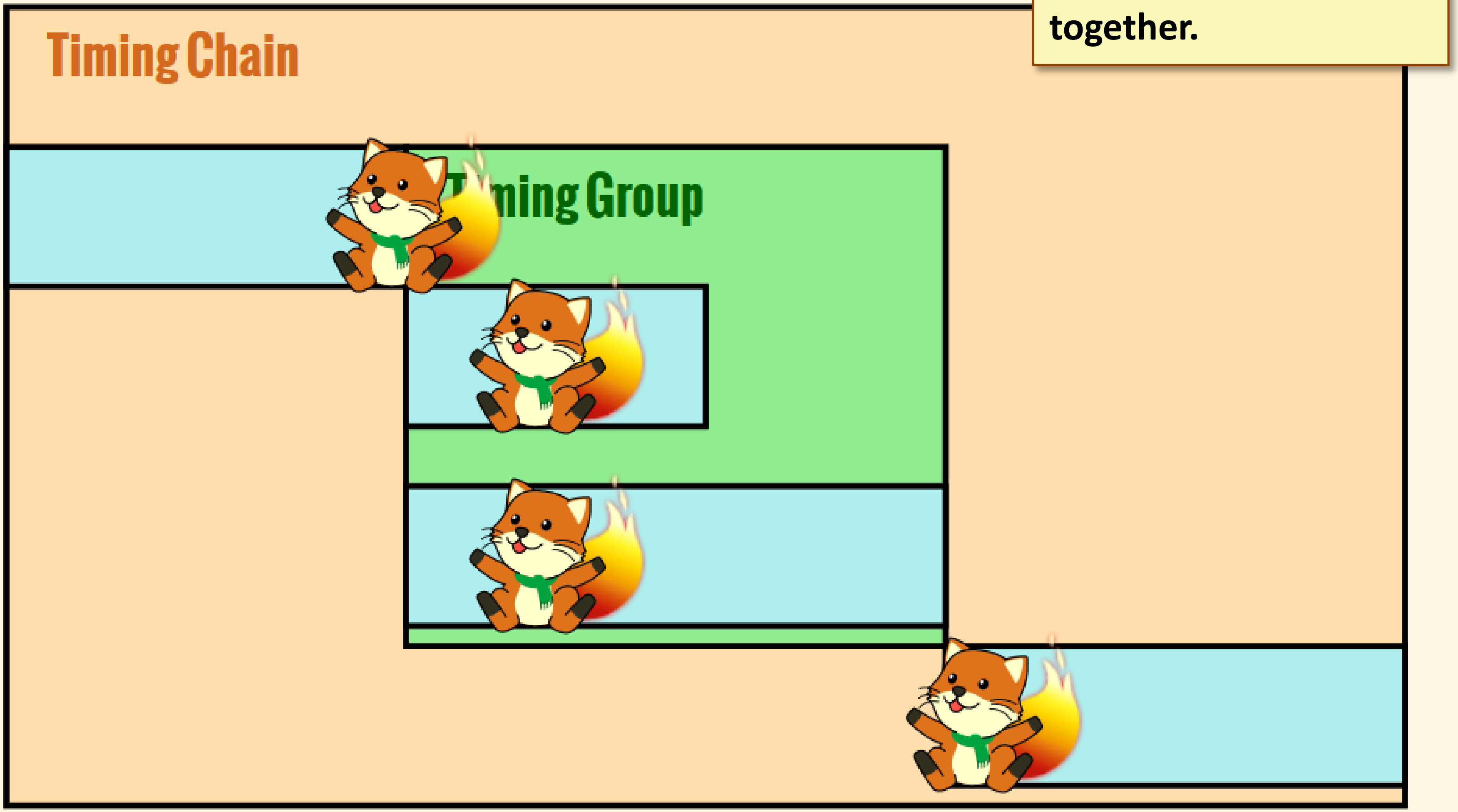
Timing groups



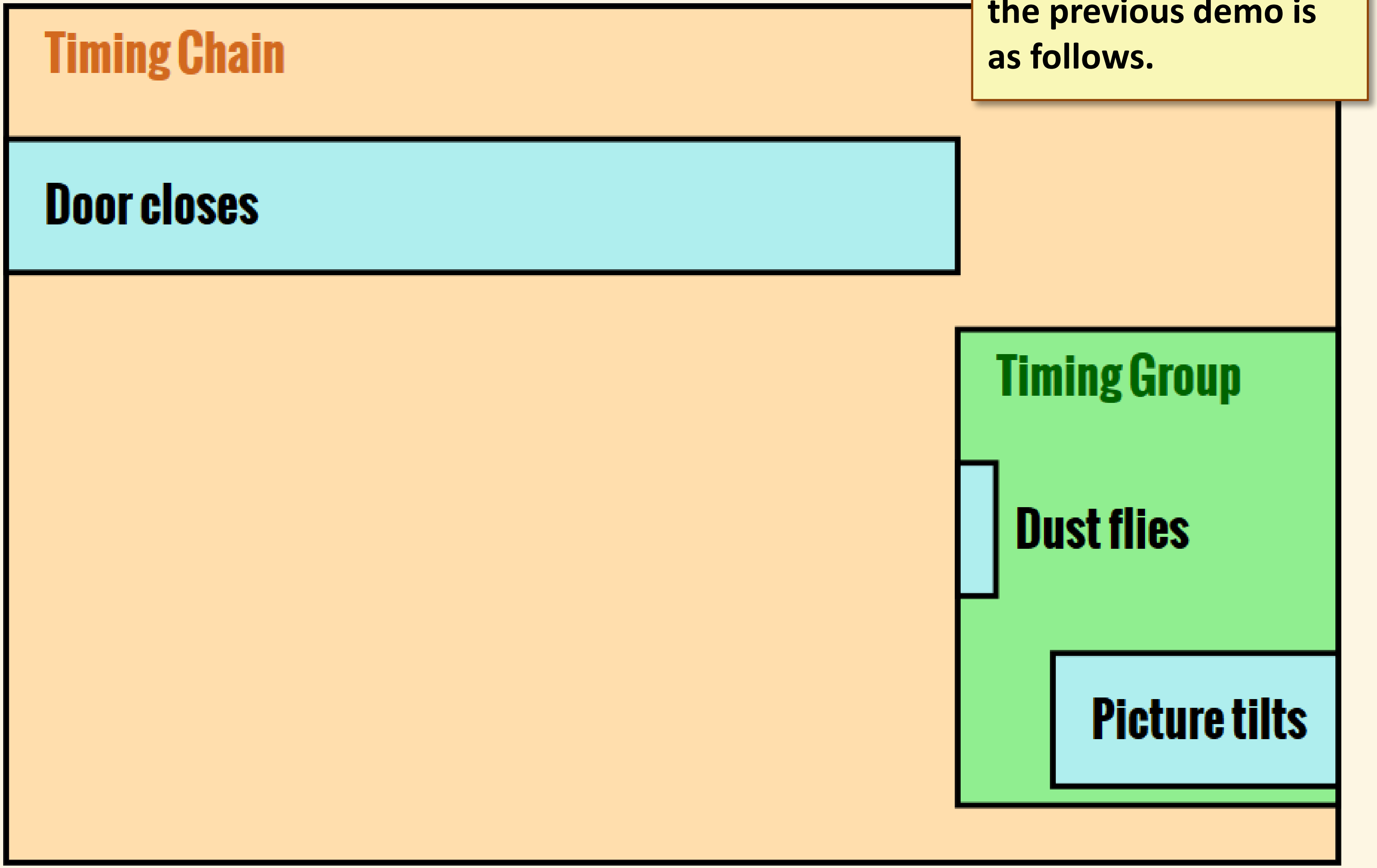
Timing chains play their children in turn.

Timing groups

You can use them together.



The arrangement from the previous demo is as follows.



Timing Chain

Door closes

Timing Group

Dust flies

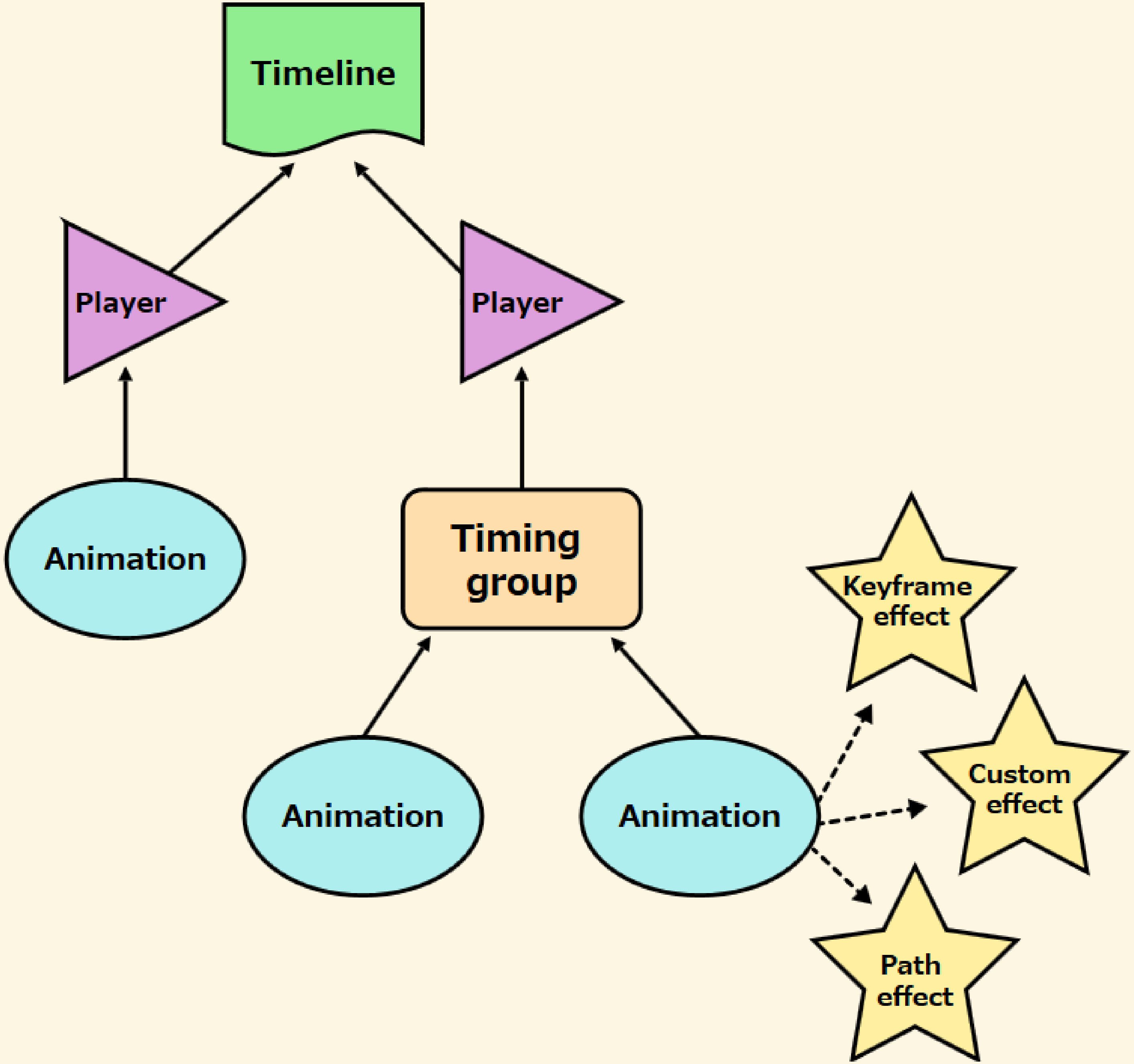
Picture tilts

Custom effect

```
function sample (
  timeFraction /* distance in interval */,
  iteration /* iteration index */,
  target /* target element */,
  previousTimeFraction)
{
    …
}
```

Path animation

```
new Animation (
  element,
  new PathAnimationEffect (
    'M 100 200 ' + … +
    'C 800 100 900 100 900 100',
    'auto-rotate'),
  { duration: 5 }
);
```


Timeline
Player
Player
Animation
Timing group
Keyframe effect
Custom effect
Animation
Animation
Path effect

Spec

dev.w3.org/fxtf/web-animations

Polyfill

git.io/webanim

Native implementation status.

Progressing

Begun

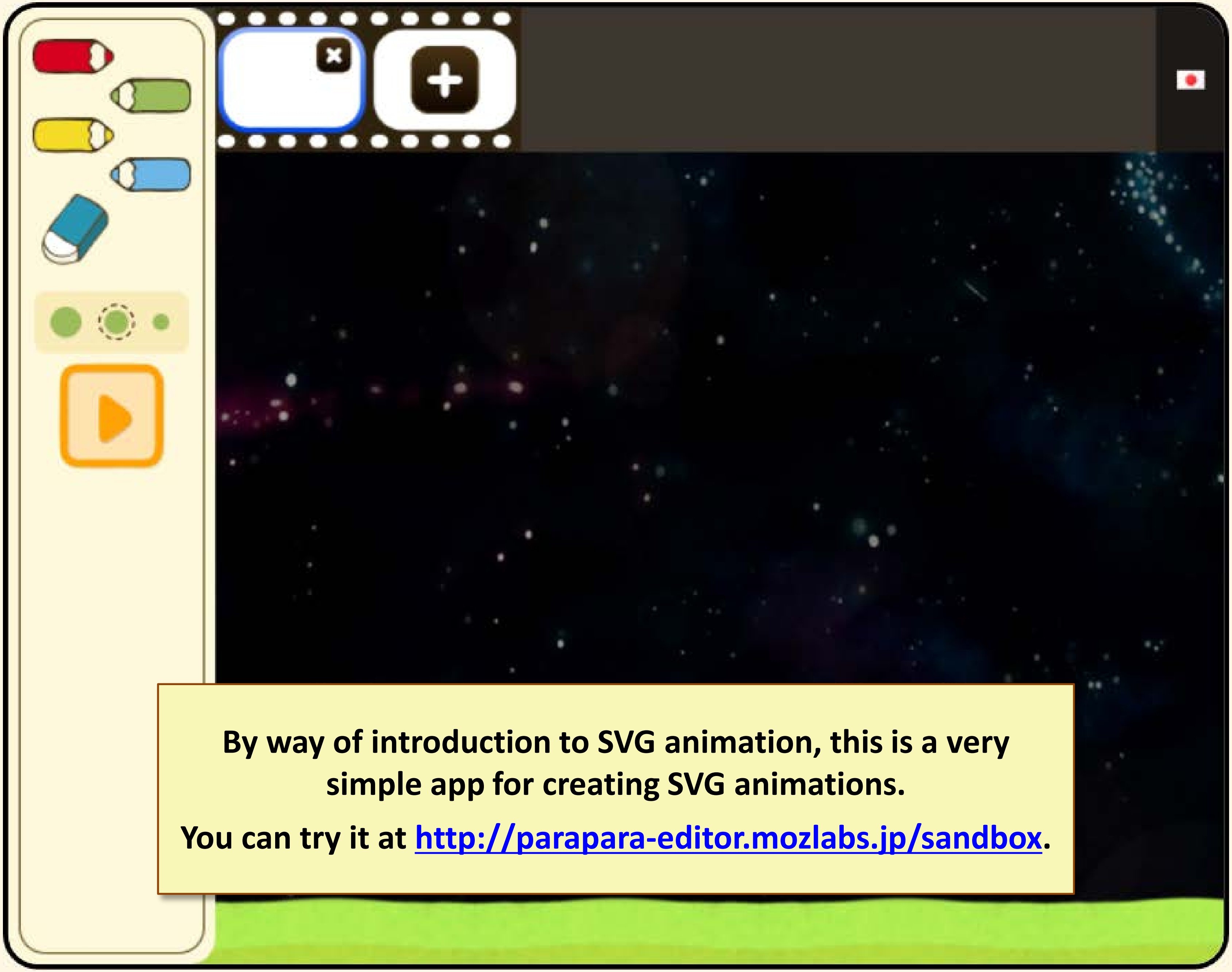
Begun

No announcement

Part 4

SVG Animation Reloaded

Animation Elements

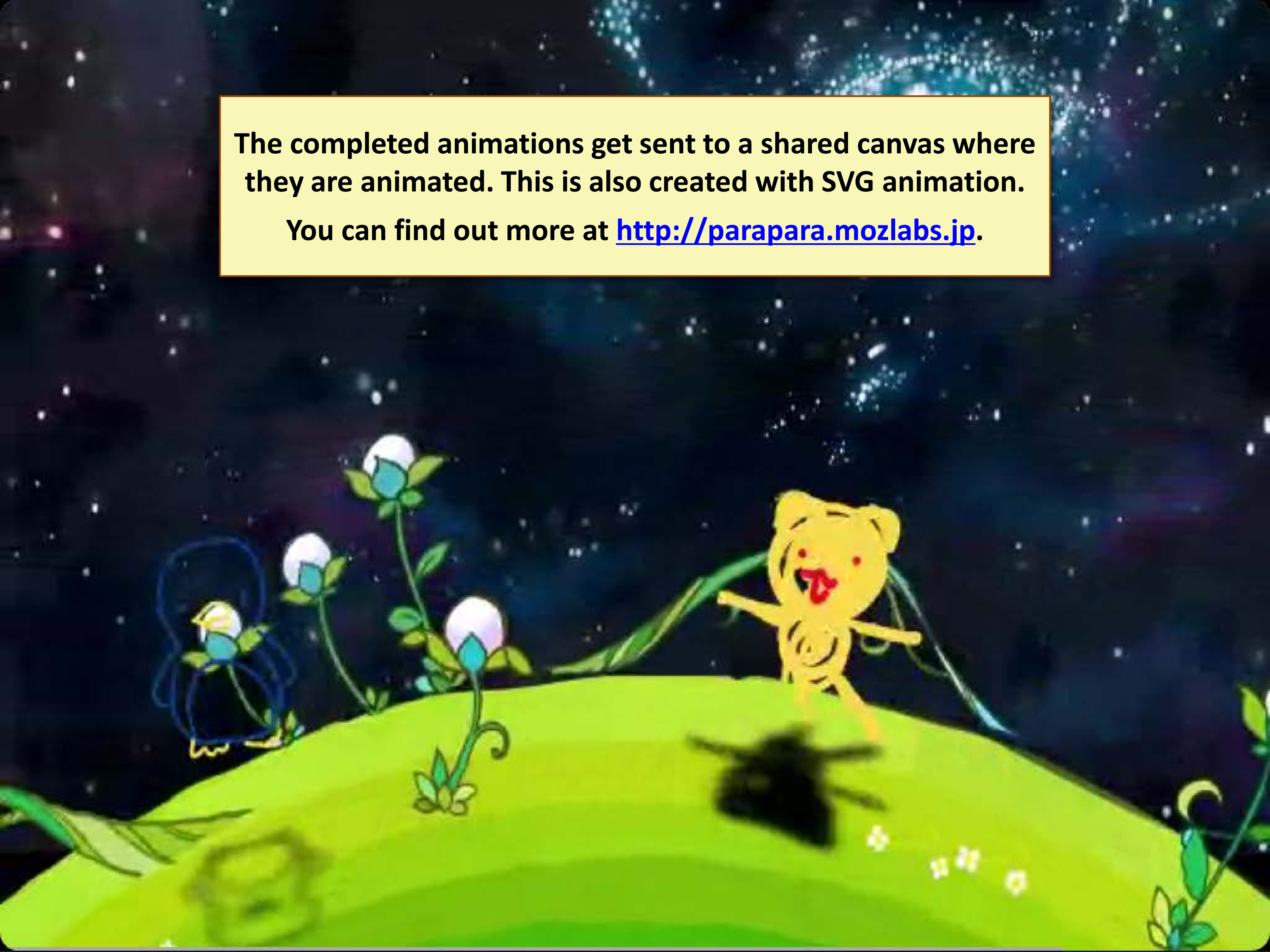


By way of introduction to SVG animation, this is a very simple app for creating SVG animations.

You can try it at http://parapara-editor.mozlabs.jp/sandbox.

The completed animations get sent to a shared canvas where they are animated. This is also created with SVG animation.

You can find out more at http://parapara.mozlabs.jp.



The SVG file for a character looks like this.

```
<g visibility="hidden">
  …Path data…
  <set id="a" attributeName="visibility"
    to="visible" dur="0.3s"
    begin="0; c.end"/>
</g>
<g visibility="hidden">
  …Path data…
  <set id="b" attributeName="visibility"
    to="visible" dur="0.3s" begin="a.end"/>
</g>
<g visibility="hidden">
  …Path data…
  <set id="c" attributeName="visibility"
    to="visible" dur="0.3s" begin="b.end"/>
</g>
```

But with Animation Elements we can simplify it to this.

```
<g visibility="hidden" class="frame">
  …Path data…
</g>
<g visibility="hidden" class="frame">
  …Path data…
</g>
<g visibility="hidden" class="frame">
  …Path data…
</g>
<timingchain>
  <set select=".frame"
    attributeName="visibility" to="visible"
    dur="0.3s"/>
</timingchain>
```

The select attribute lets us re-use the animation definition.

```
<g visibility="hidden" class="frame">
  …Path data…
</g>
<g visibility="hidden" class="frame">
  …Path data…
</g>
<g visibility="hidden" class="frame">
  …Path data…
</g>
<timingchain>
  <set select=".frame"
    attributeName="visibility" to="visible"
    dur="0.3s"/>
</timingchain>
```

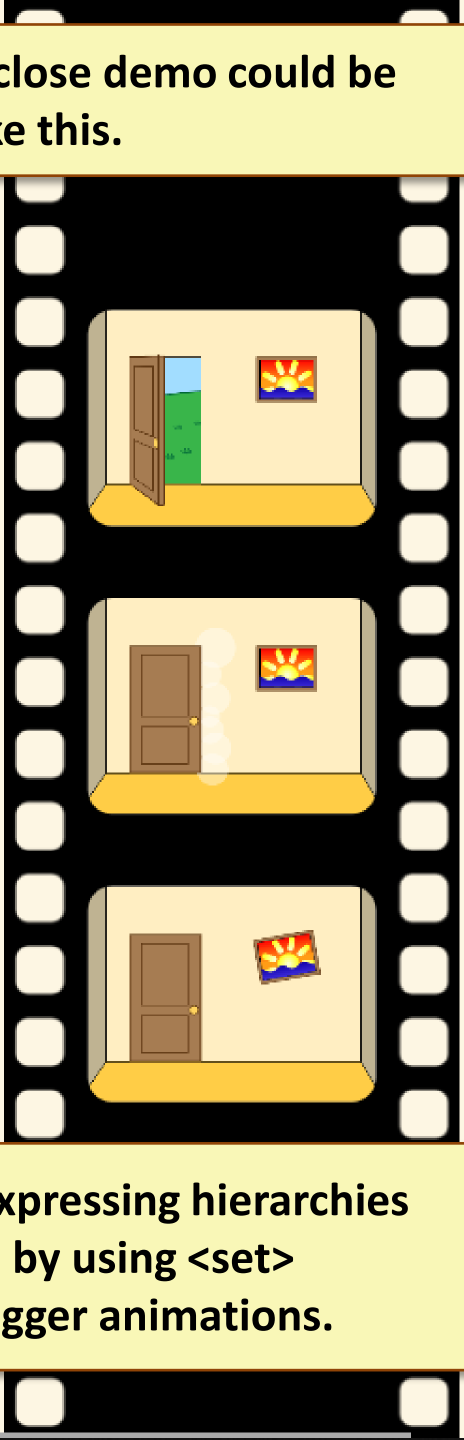
Timing chains make sequencing easy.

```
<g visibility="hidden" class="frame">
  …Path data…
</g>
<g visibility="hidden" class="frame">
  …Path data…
</g>
<g visibility="hidden" class="frame">
  …Path data…
</g>
<timingchain>
  <set select=".frame"
    attributeName="visibility" to="visible"
    dur="0.3s"/>
</timingchain>
```

Our door close demo could be written like this.

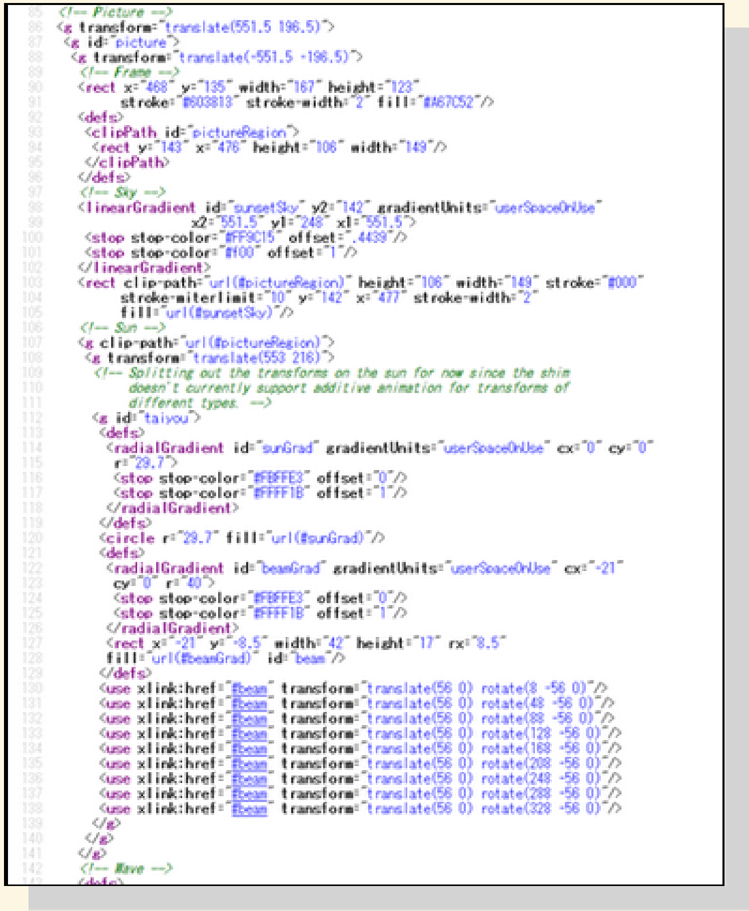
```
<timingchain>
  <!-- Close door -->
  <animate href="door"
    attributeName="transform"
    to="rotate(0deg)" dur="2s"
    easing="cubic-bezier(0.9,0,1,1)"/>
  <timinggroup>
    <!-- Tilt picture -->
    <animate href="picture"
      attributeName="transform"
      to="rotate(-10deg)" dur="0.15s"/>
    <!-- Scatter dust -->
    <animate href="smoke"
      attributeName="opacity"
      values="0; 0.8; 0" dur="0.2s"
      begin="0.4s"/>
 </timinggroup>
</timingchain>
```

Element syntax is convenient for expressing hierarchies and can be used together with CSS by using <set> elements to set CSS classes that trigger animations.



This kind of approach lends itself to collecting timing information in one place, perhaps even a separate timesheet file.

Graphical content



```
<!-- Picture -->
<g transform="translate(551.5 196.5)">
 <g id="picture">
  <g transform="translate(-551.5 -196.5)">
   <!-- Frame -->
   <rect x="468" y="135" width="167" height="123"
         stroke="#603813" stroke-width="2" fill="#A67C52"/>
   <defs>
    <clipPath id="pictureRegion">
     <rect y="143" x="476" height="106" width="149"/>
    </clipPath>
   </defs>
   <!-- Sky -->
   <linearGradient id="sunsetSky" y2="142" gradientUnits="userSpaceOnUse"
                   x2="551.5" y1="248" x1="551.5">
    <stop stop-color="#FF9C15" offset=".4439"/>
    <stop stop-color="#f00" offset="1"/>
   </linearGradient>
   <rect clip-path="url(#pictureRegion)" height="106" width="149" stroke="#000"
         stroke-miterlimit="10" y="142" x="477" stroke-width="2"
         fill="url(#sunsetSky)"/>
   <!-- Sun -->
   <g clip-path="url(#pictureRegion)">
    <g transform="translate(553 216)">
     <!-- Splitting out the transforms on the sun for now since the shim
          doesn't currently support additive animation for transforms of
          different types. -->
     <g id="taiyou">
      <defs>
       <radialGradient id="sunGrad" gradientUnits="userSpaceOnUse" cx="0" cy="0"
        r="29.7">
        <stop stop-color="#FBFFE3" offset="0"/>
        <stop stop-color="#FFFF1B" offset="1"/>
       </radialGradient>
      </defs>
      <circle r="29.7" fill="url(#sunGrad)"/>
      <defs>
       <radialGradient id="beamGrad" gradientUnits="userSpaceOnUse" cx="-21"
        cy="0" r="40">
        <stop stop-color="#FBFFE3" offset="0"/>
        <stop stop-color="#FFFF1B" offset="1"/>
       </radialGradient>
       <rect x="-21" y="-8.5" width="42" height="17" rx="8.5"
       fill="url(#beamGrad)" id="beam"/>
      </defs>
      <use xlink:href="#beam" transform="translate(56 0) rotate(8 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(48 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(88 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(128 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(168 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(208 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(248 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(288 -56 0)"/>
      <use xlink:href="#beam" transform="translate(56 0) rotate(328 -56 0)"/>
     </g>
    </g>
   </g>
   <!-- Wave -->
   <defs>
```

Time sheet

Polyfill

git.io/webanim

Standards discussion

public-fx@w3.org

Brian Birtles

birtles@mozilla-japan.org