

## BASIC OBJECTS

**circle** (center, size, color)  
**cone** (center, size, color)  
**cube** (center, size, color)  
**cylinder** (center, size, color)  
**line** (center, size, color)  
**point** (center, size, color)  
**polygon** (count, center, size, color)  
**prism** (count, center, size, color)  
**pyramid** (count, center, size, color)  
**sphere** (center, size, color)  
**square** (center, size, color)

## BASIC PROPERTIES

**.center** = [x, y, z]  
**.color** = "colorname" / 0xFFFFFF / [r,g,b]<sup>0...1</sup>  
= **rgb** (r,g,b)<sup>255</sup> / **hsl** (h<sup>360</sup>,s<sup>100</sup>,l<sup>100</sup>)  
**.count** = count / [count, count]  
**.image** = drawing / "filename" / **image** ("filename")  
**.images** = .count / [count, count]  
**.size** = width / [.width, .height, .depth]  
**.spin** = spinH / [.spinH, .spinV, .spinT]  
**.visible** / **.hidden** = true / yes / false / no  
**.wireframe** = true / yes / false / no

## SUICA

**background** (color)  
**xyz** (size, color)  
**demo** (distance, altitude, speed)  
**orbit** (distance, altitude, speed)  
**lookAt** (from, to, up)  
**perspective** (near, far, fov)  
**orthographic** (near, far)  
**fullWindow** ()  
**fullScreen** ()  
**stereo** (distance)  
**anaglyph** (distance)  
**vr** ()  
**capture** (filename, time, fps, format, skipframes)

## MISC

**its**  
**obj.clone**  
**obj.style** ({name: value, ...})  
**allObjects** ()  
**findPosition** (event)  
**findObject** (event)  
**findObjects** (event)  
**objectPosition** (local)  
**screenPosition** (local, global)  
**radians** (degrees)  
**degrees** (radians)  
**random** (from, to)  
**random** ([value,...])

## ADVANCED OBJECTS

**construct** (expression, size, color)  
**convex** (src<sup>2</sup>, size, color)  
**group** (object, object, ...)  
    **.add** (object, object, ...)  
**model** (filename, center, size)  
**model.save** (filename, [object, object, ...])  
**surface** (center, curve<sup>3</sup>, count, size, color)  
**text3d** (text, font<sup>1</sup>, center, size, color)  
**tube** (center, curve<sup>2</sup>, radius, count, size, color)  
**spline** (src<sup>2</sup>, closed, interpolating)  
    (src<sup>5</sup>, param, param)  
**spplane** (src<sup>3</sup>, closed<sup>1,2</sup>, interpolating<sup>1,2</sup>)  
    (src<sup>4</sup>, param, param)

## ADVANCED PROPERTIES

**.closed** = bool<sup>1</sup> / [bool, bool]<sup>2</sup>  
**.curve**<sup>2</sup> = [point, ...] / spline / f(u)  
    <sup>3</sup> = [[point, ...], ...] / spline / f(u,v)  
**.expression** = "string" A+B, A-B, A\*B, (...)  
**.font**<sup>1</sup> = "fontname.json"  
**.interpolating** = bool<sup>1</sup> / [bool, bool]<sup>2</sup>  
**.src** = [point, ... point]<sup>2</sup> / f(u)<sup>5</sup>  
    = [[point, ...], ... [point, ...]]<sup>3</sup> / f(u,v)<sup>4</sup>  
**.vertices**  
**.threejs** = THREE.Mesh  
    **.material** = THREE.Material  
    **.geometry** = THREE.BufferGeometry

## EVENTS

**onPointerEnter, onPointerLeave, onPointerMove,**  
**onPointerDown, onPointerUp, onClick, onTime,**  
**onLoad**

**obj.addEventListener** (eventName, eventHandler)  
**obj.removeEventListener** (eventName)  
**obj.eventName** = eventHandler

**function** pointerEventHandler (event) { ... }  
**function** timeEventHandler (time, dTime) { ... }  
**function** loadEventHandler (object) { ... }

**proactive** ()

## DRAWINGS

**drawing** (width, height, color)  
**moveTo** (x, y, x, y, ...)  
**lineTo** (x, y, x, y, ...)  
**curveTo** (m<sub>x</sub>, m<sub>y</sub>, x, y)  
**arc** (x, y, radius, from, to, cw)  
**stroke** (color, width, closed)  
**fill** (color)  
**fillText** (x, y, text, color, font)  
    "bold 20px Courier"  
**clear** (color)

## LMS

**scorm**  
**.api, .score, .studentName, .getValue** (value) **.setValue** (name, value), **.derandomize** (seed)

# Suica 2.0 for HTML

## BASIC OBJECTS

```
<circle center size color ...>
<cone center size color ...>
<cube center size color ...>
<cylinder center size color ...>
<line from to color ...>
<point center size color ...>
<polygon count center size color ...>
<prism count center size color ...>
<pyramid count center size color ...>
<sphere center size color ...>
<square center size color ...>
    <... id spin image images wireframe>
```

## ADVANCED OBJECTS

```
<clone src1 center size color ...>
<construct expression center size color>
<convex src2 size color ...>
<group center size color ...> ... </group>
<model filename center size ...>
<surface center curve3 count size color ...>
<text3d text font1 center size color ...>
<tube center curve2 radius count size color ...>
<splane src2,5 closed1 interpolating1>
<splane src3,4 closed1,2 interpolating1,2>
    <... id spin image images wireframe>
```

```
<script src="suica.js"></script>
```

## BASIC PROPERTIES

```
center = "x, y, z"
color = "colorname" / "0xFFFFFF", "r,g,b"
    = "rgb (r,g,b)"255 / "hsl (h360,s100,l100)"
count = "count" / "count, count"
id = "string"
image = "drawing" / "filename"
images = "count" / "count, count"
size = "width" / "width, height, depth"
spin = "spinH" / "spinH, spinV, spinT"
visible / hidden = "bool"
wireframe = "bool"
```

## ADVANCED PROPERTIES

```
closed = "bool"1 / "bool, bool"2
curve2 = "point; ..." / "spline" / "func(u)"
3 = "point; ... | ..." / "splane" / "func(u,v)"
expression = "string" A+B, A-B, A*B, (...)
font1 = "fontname.json"
interpolating = "bool"1 / "bool, bool"2
src = "id" / "point; ..." / "func(u)"5
    = "point; ... | ..." / "func(u,v)"4

interpolating vs approximating
closed vs open, cw vs ccw
```

## SUICA

```
<suica width height background orientation proactive
    perspective orthographic fullWindow fullScreen
    stereo anaglyph vr> ... </suica>
<background color>
<oxyz size color>
<demo distance altitude speed>
<orbit id distance altitude speed>
<lookAt from to up>
<perspective near far fov>
<orthographic near far>
<fullWindow>
<fullScreen>
<stereo distance>
<anaglyph distance>
<vr>
<capture filename time fps format skipframes>
```

## EVENTS

```
onPointerEnter, onPointerLeave, onPointerMove,
onPointerDown, onPointerUp, onClick, onTime, onLoad

<tag ... eventName="eventHandler">

<proactive>
```

[boytchev.github.io/suica](https://github.com/boytchev/suica)

version 1

## DRAWINGS

```
<drawing size color>
<moveTo point>
<lineTo point>
<curveTo m point>
<arc point radius from to cw>
<stroke color width closed>
<fill color>
<fillText point text color font>
<clear color>
```

```
point="x, y" or x="x" y="y"
font="bold 20px Courier"
```