#WWDC18

Integrating Apps and Content with AR Quick Look

Session 603

David Lui, ARKit Engineering
Dave Addey, ARKit Engineering

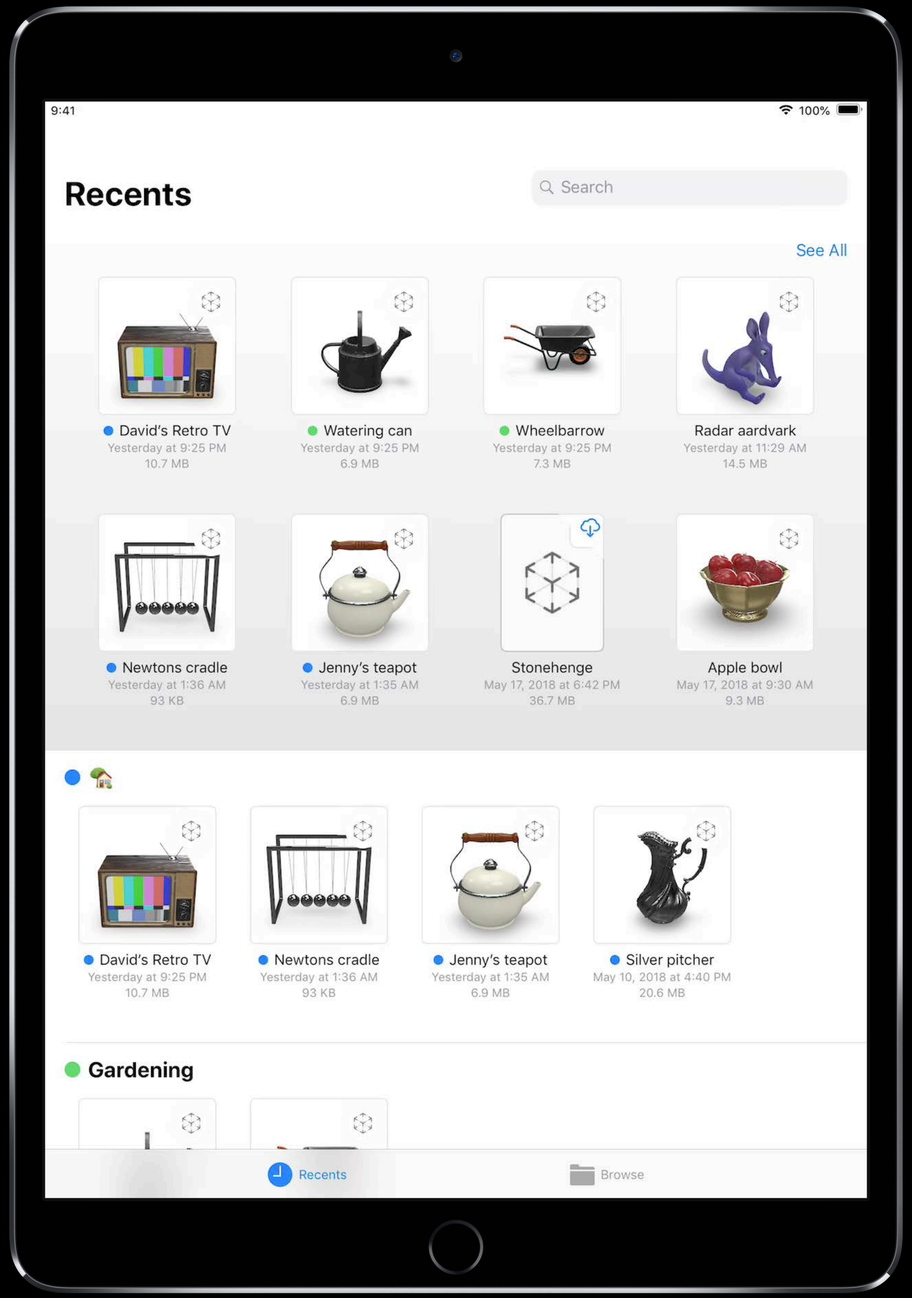
Overview

What is AR Quick Look?

Adopting AR Quick Look

Creating 3D models for AR Quick Look

What Is AR Quick Look?


9:41
100%
Recents
Search
See All
David's Retro TV
Yesterday at 9:25 PM
10.7 MB
Watering can
Yesterday at 9:25 PM
6.9 MB
Wheelbarrow
Yesterday at 9:25 PM
7.3 MB
Radar aardvark
Yesterday at 11:29 AM
14.5 MB
Newtons cradle
Yesterday at 1:36 AM
93 KB
Jenny's teapot
Yesterday at 1:35 AM
6.9 MB
Stonehenge
May 17, 2018 at 6:42 PM
36.7 MB
Apple bowl
May 17, 2018 at 9:30 AM
9.3 MB
David's Retro TV
Yesterday at 9:25 PM
10.7 MB
Newtons cradle
Yesterday at 1:36 AM
93 KB
Jenny's teapot
Yesterday at 1:35 AM
6.9 MB
Silver pitcher
May 10, 2018 at 4:40 PM
20.6 MB
Gardening
Recents
Browse

What Is AR Quick Look?

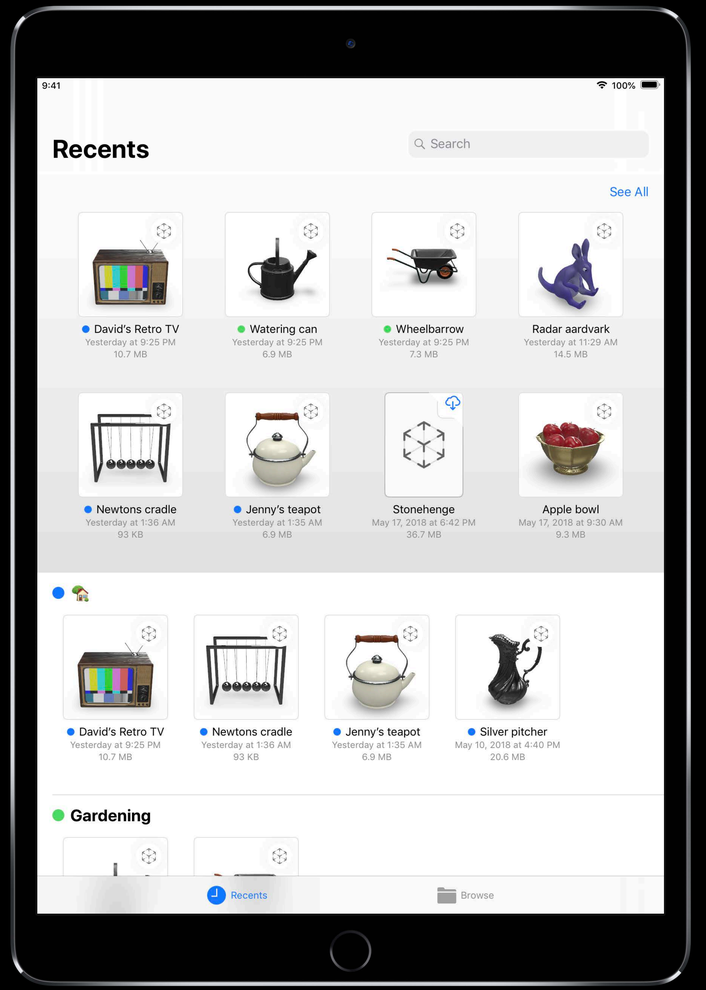
NEW

Preview 3D content in AR

System-wide

Uses Quick Look technology

Available on iOS 12



Realistic rendering

AR onboarding

Peek and Pop

Environment Texturing

AR Snapshots

Plane detection

Object placement with device movement

Haptics

Translation gestures

Rotation manipulation

Lighting

100% scale viewing

Object scaling

Contact shadows

Skeletal animation

Transform animation

Hit-testing

File sharing

VoiceOver

Switch Control

Developer Benefits

No need to create your own

Really easy to integrate

Built-in AR setup

No understanding of AR technology required

Demo


AR

‘usdz’ File

New file format for distributing

Packages a model and its textu file

Based on Pixar’s open-source U

Supported on iOS and macOS

`usdz_converter` tool in Xcode 10

Adopting AR Quick Look

AR Quick Look Integration

Apps

Website

AR Quick Look Integration

Apps

Website

Quick Look

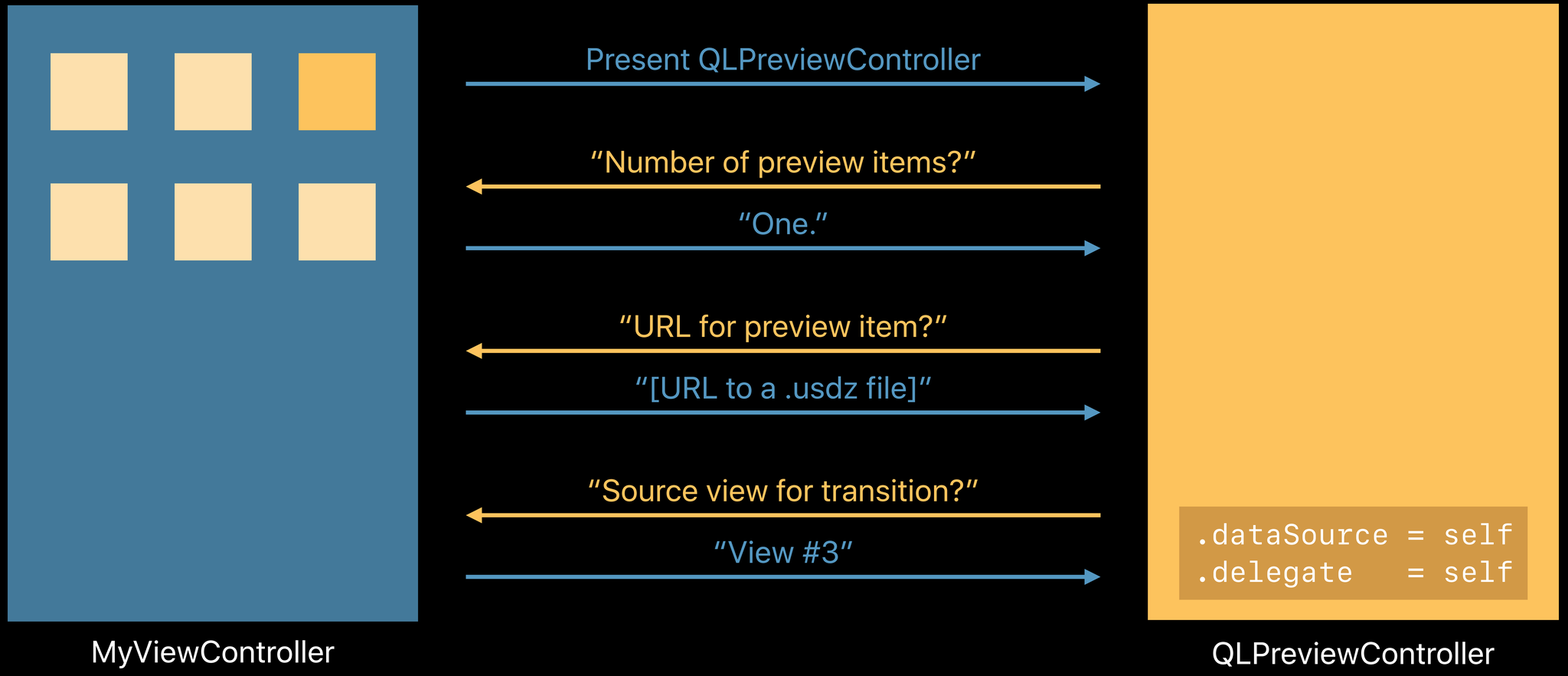
Preview documents

Framework for previewing file formats

Control over transitions and presentation modes

Secure and private

Quick Look Preview Flow



Previewing ‘usdz’ Objects Using Quick Look

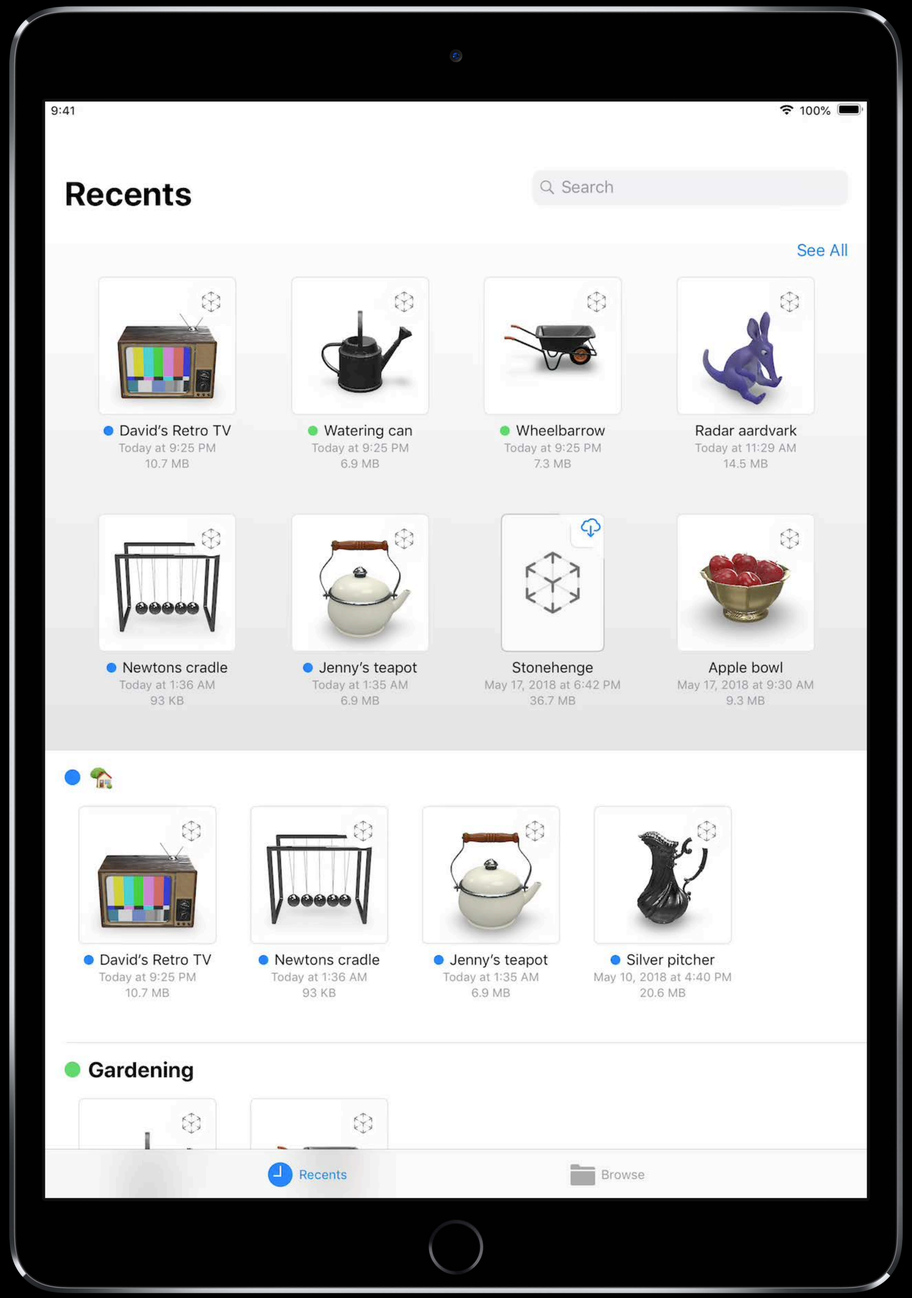
QLPreviewController

```
func preview(_ sender: Any) {
        let previewController = QLPreviewController()
        previewController.dataSource = self
        previewController.delegate = self

        // Present viewer modally
        present(previewController, animated: true, completion: nil)
}
```

Previewing ‘usdz’ Objects Using Quick Look

QLPreviewControllerDataSource

```
func numberOfPreviewItems(in controller: QLPreviewController) -> Int {
        // Viewer supports previewing a single 3D object
        return 1
}

func previewController(
  _ controller: QLPreviewController, previewItemAt index: Int) -> QLPreviewItem {
        // Return the file URL to the .usdz file
        let fileUrl = Bundle.main.url(forResource: “radar_aardvark”, withExtension: “usdz”)!
        return fileUrl as QLPreviewItem
}
```


9:41
100%
Recents
Search
See All
David's Retro TV
Today at 9:25 PM
10.7 MB
Watering can
Today at 9:25 PM
6.9 MB
Wheelbarrow
Today at 9:25 PM
7.3 MB
Radar aardvark
Today at 11:29 AM
14.5 MB
Newtons cradle
Today at 1:36 AM
93 KB
Jenny's teapot
Today at 1:35 AM
6.9 MB
Stonehenge
May 17, 2018 at 6:42 PM
36.7 MB
Apple bowl
May 17, 2018 at 9:30 AM
9.3 MB
David's Retro TV
Today at 9:25 PM
10.7 MB
Newtons cradle
Today at 1:36 AM
93 KB
Jenny's teapot
Today at 1:35 AM
6.9 MB
Silver pitcher
May 10, 2018 at 4:40 PM
20.6 MB
Gardening
Recents
Browse

Previewing ‘usdz’ Objects Using Quick Look

QLPreviewControllerDelegate

```
func previewController(_ controller: QLPreviewController, transitionViewFor item:
QLPreviewItem) -> UIView? {
        // Provide the starting view for a seamless zoom transition to the viewer
        return self.startingZoomView
}
```

AR Quick Look Integration

Apps

Website

AR Quick Look Integration

Apps

Website

Previewing ‘usdz’ Content in Safari

Recommended image-based experience

HTML markup

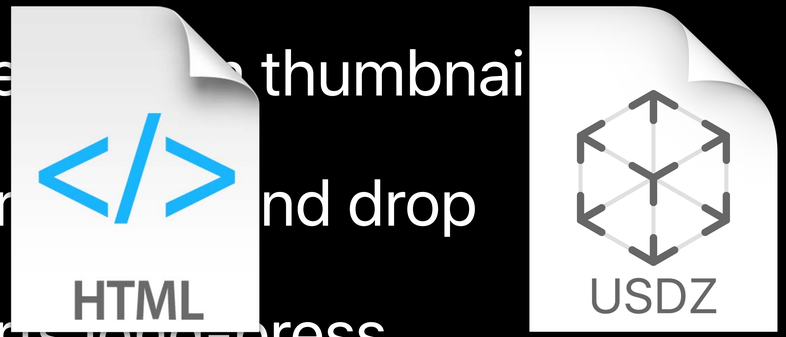
Automatic badging

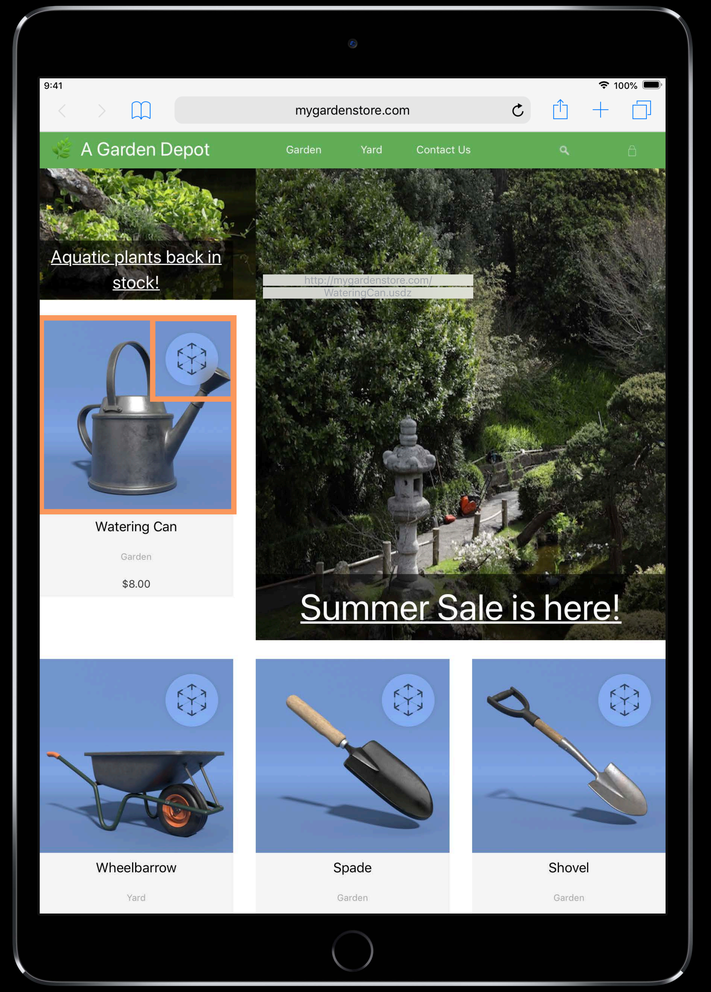
Provide thumbnai

Suppor nd drop

Supports long-press

Better workflow

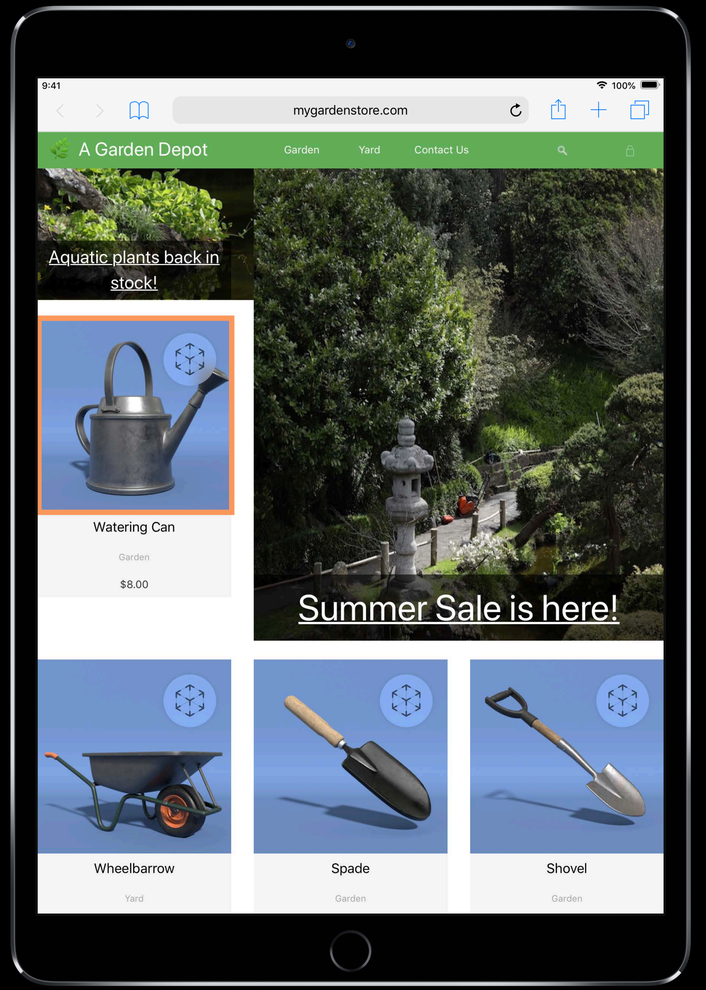




HTML Markup for Previewing ‘usdz’ Objects

<img> element

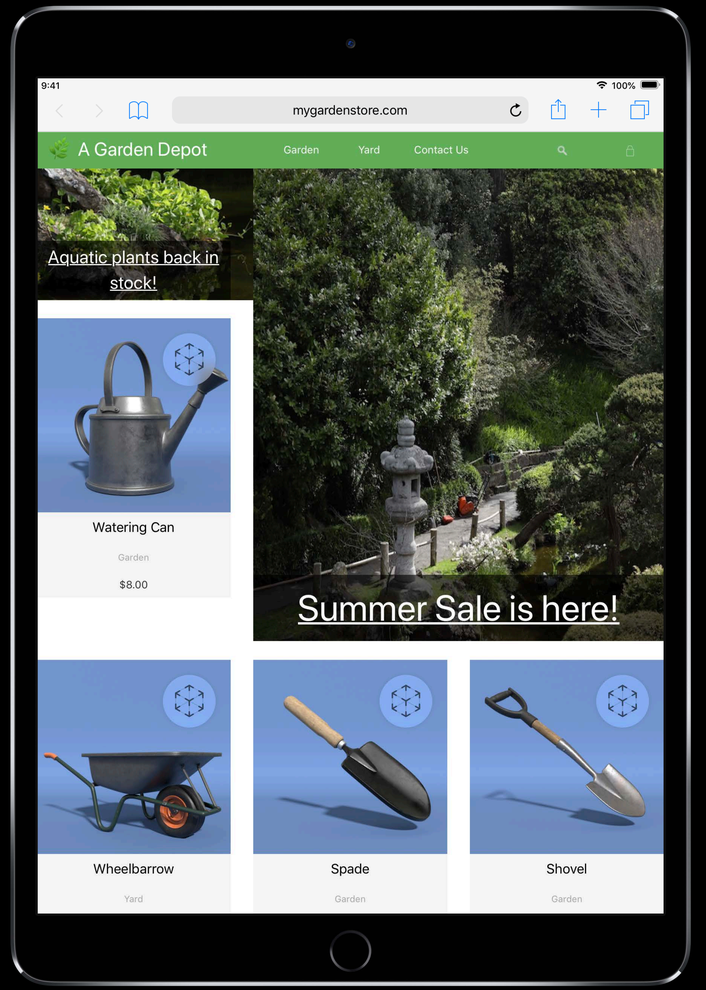
```
<a rel="ar" href="model.usdz">
  <img src="model-preview.jpg">
</a>
```



HTML Markup for Previewing ‘usdz’ Objects

<picture> element

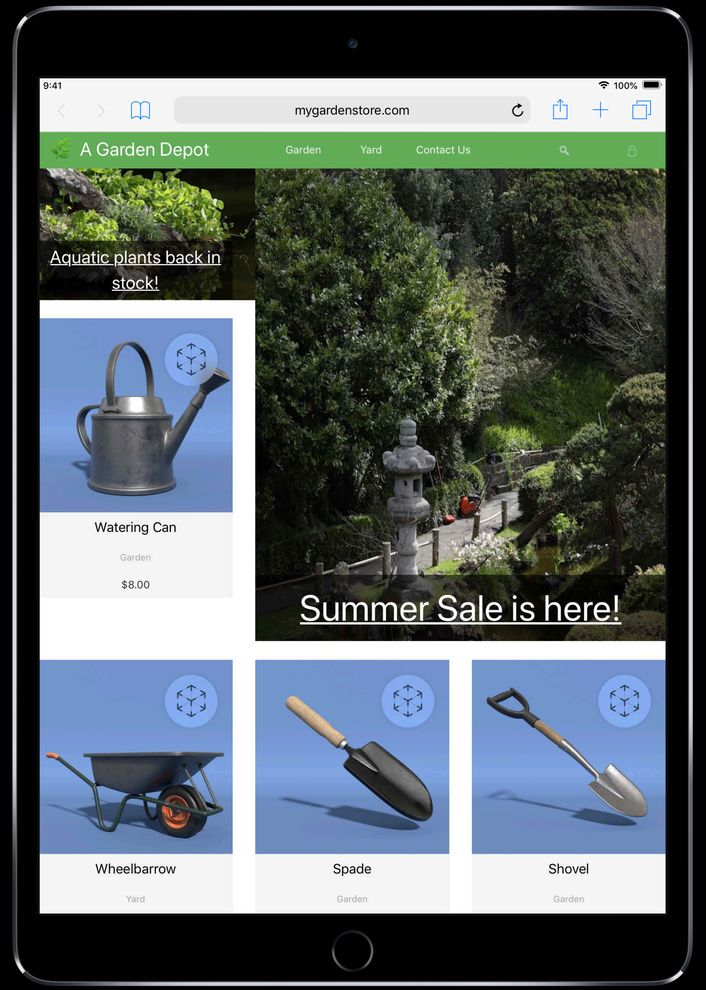
```
<a rel="ar" href="model.usdz">
  <picture>
    <source srcset=“wide-image.png”
        media="(min-width: 600px)">
    <img src="narrow-image.png">
  </picture>
</a>
```



HTML Markup for Previewing ‘usdz’ Objects

MIME type

```
AddType model/vnd.pixar.usd .usdz
AddType model/usd usdz
```



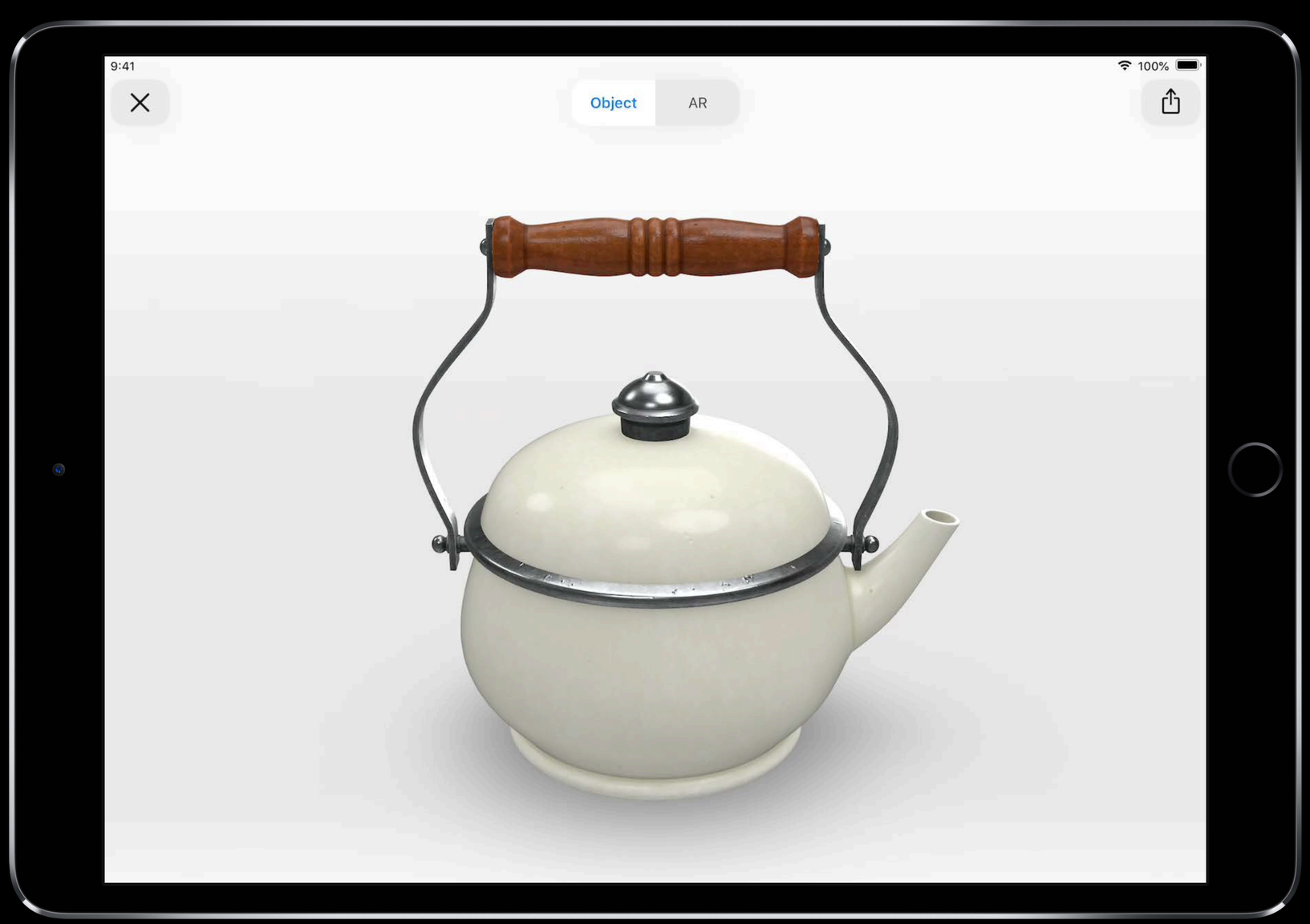
AR Quick Look Integration

Apps

Website

Creating 3D Models for AR Quick Look

Dave Addey, ARKit Engineering


9:41
100%
Object
AR

Creating 3D Models

Placement

Physical size

Animation

Contact shadow

Appearance

Transparency

Optimizing and exporting models

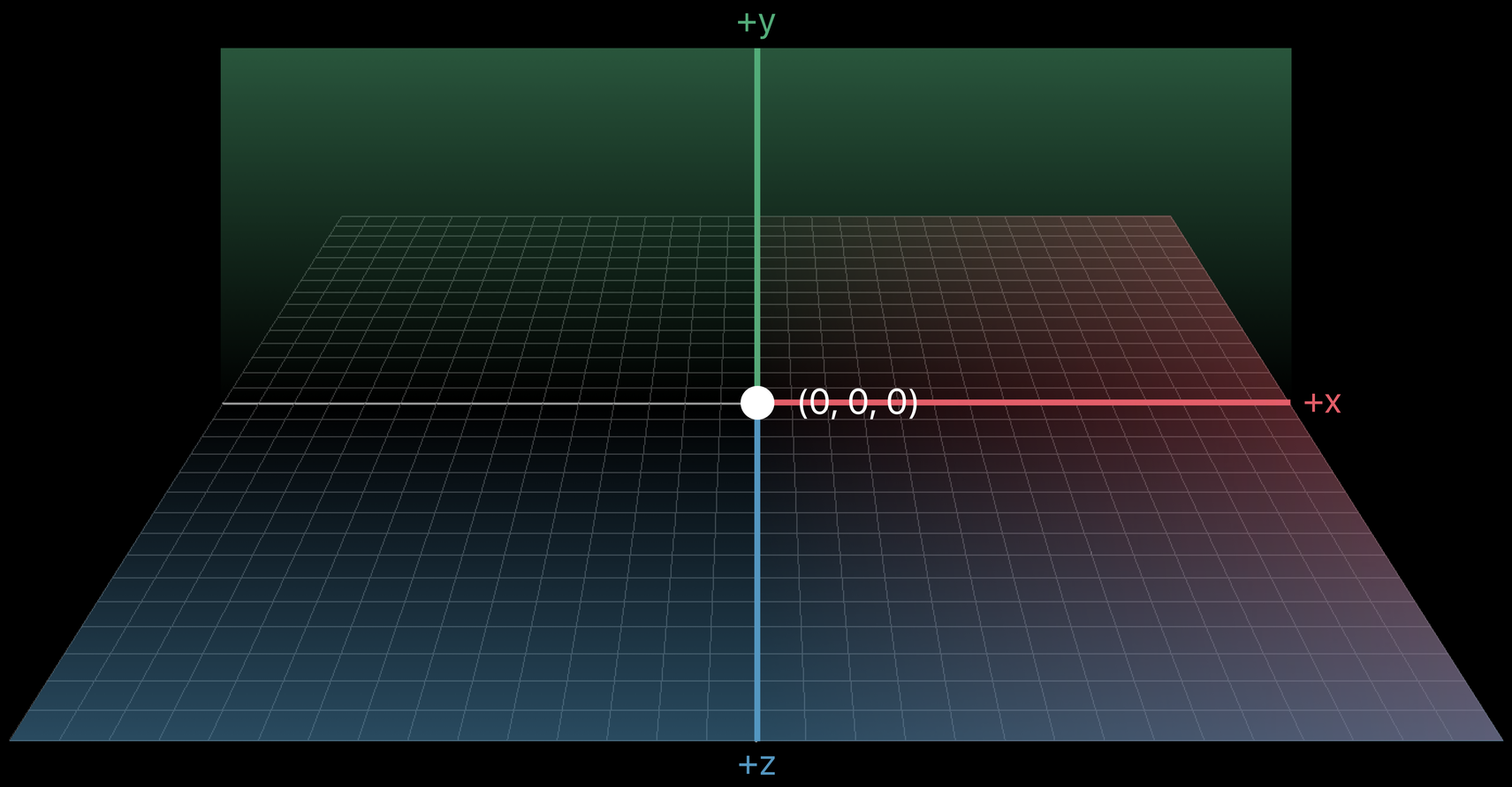
Placement

Place objects facing towards the camera (facing +z)

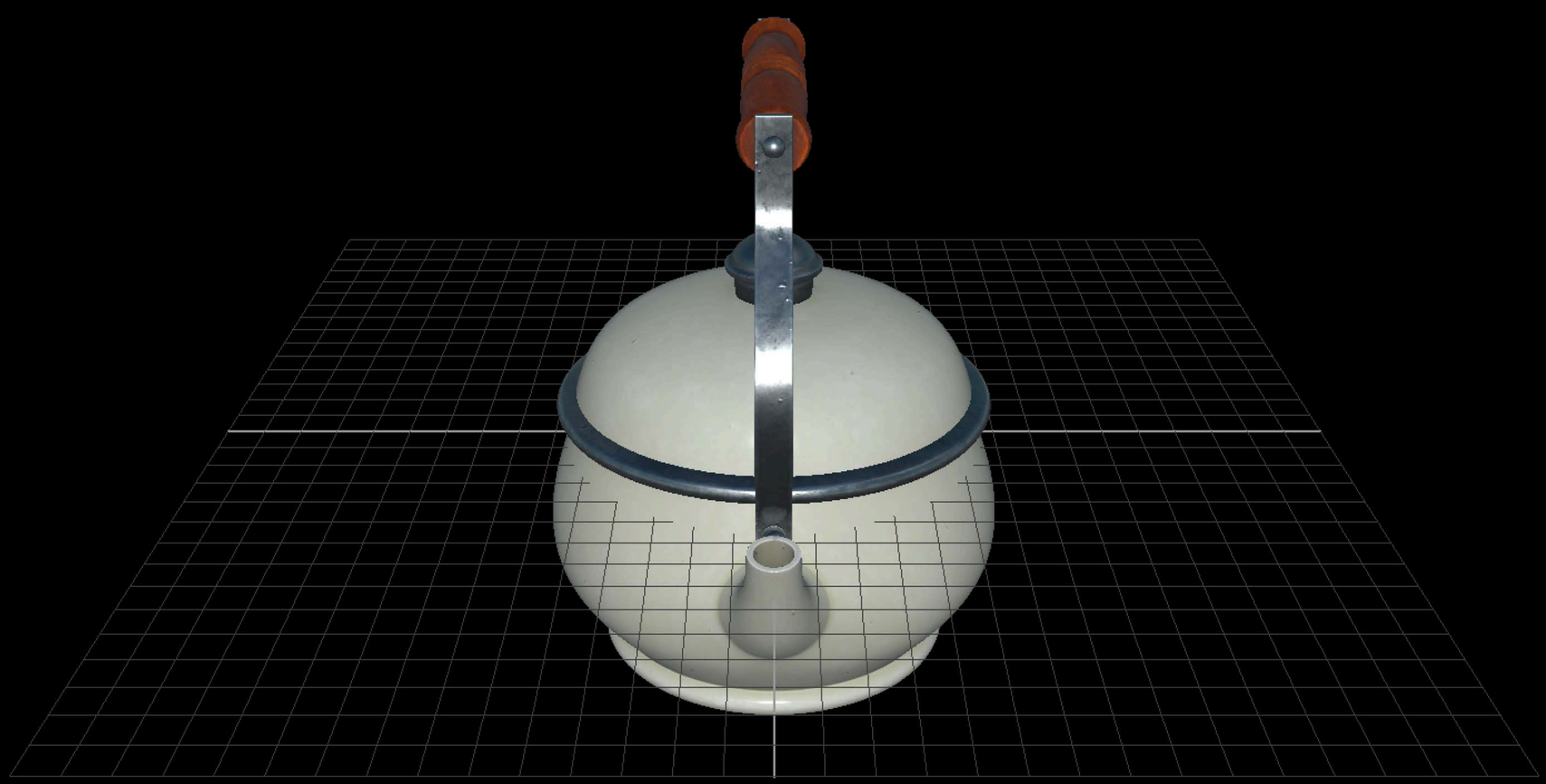
Base of object should sit on the ground plane (y = 0)

Pivot point should be at the origin (x, y, z = 0)

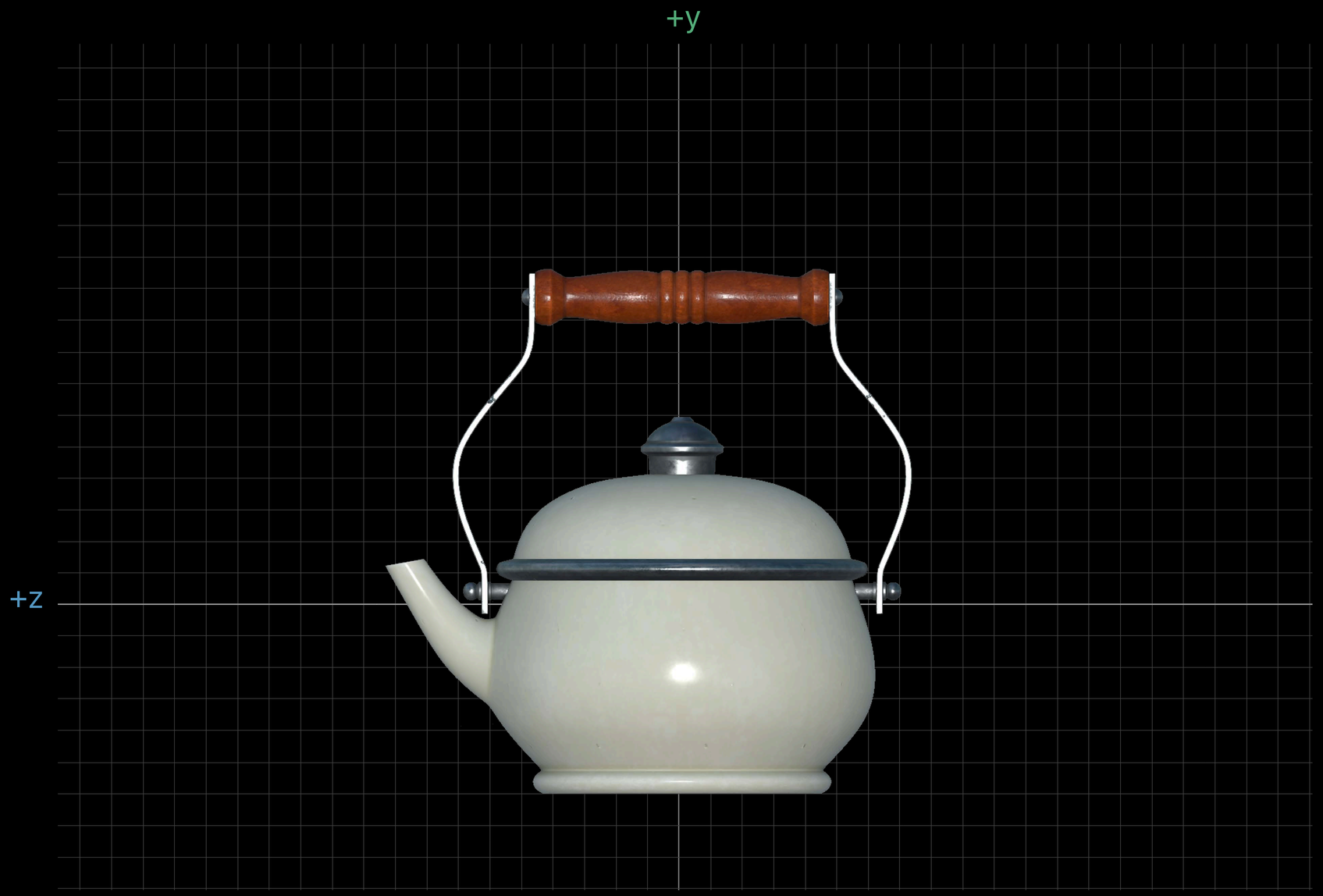
Placement



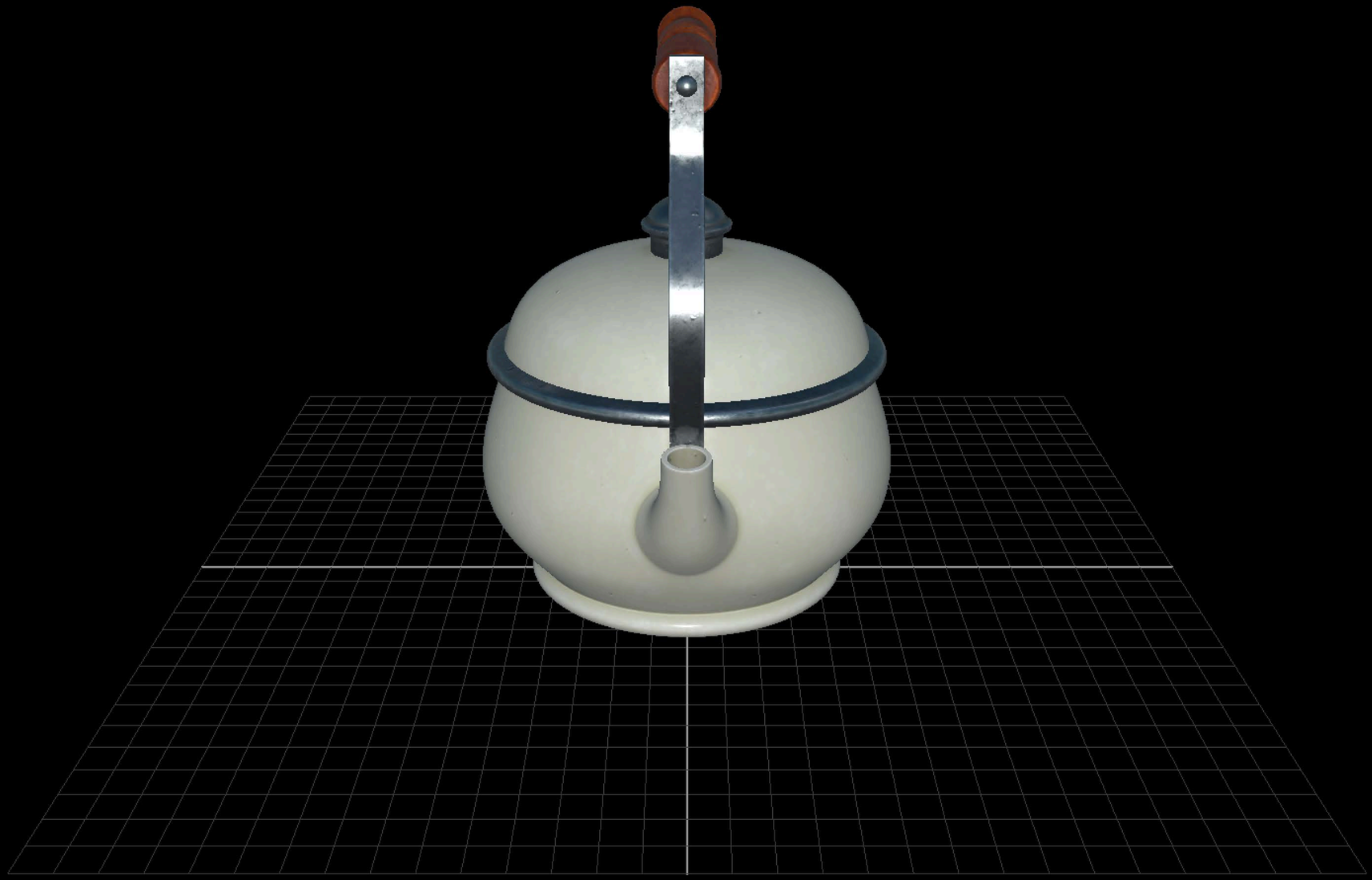
Placement



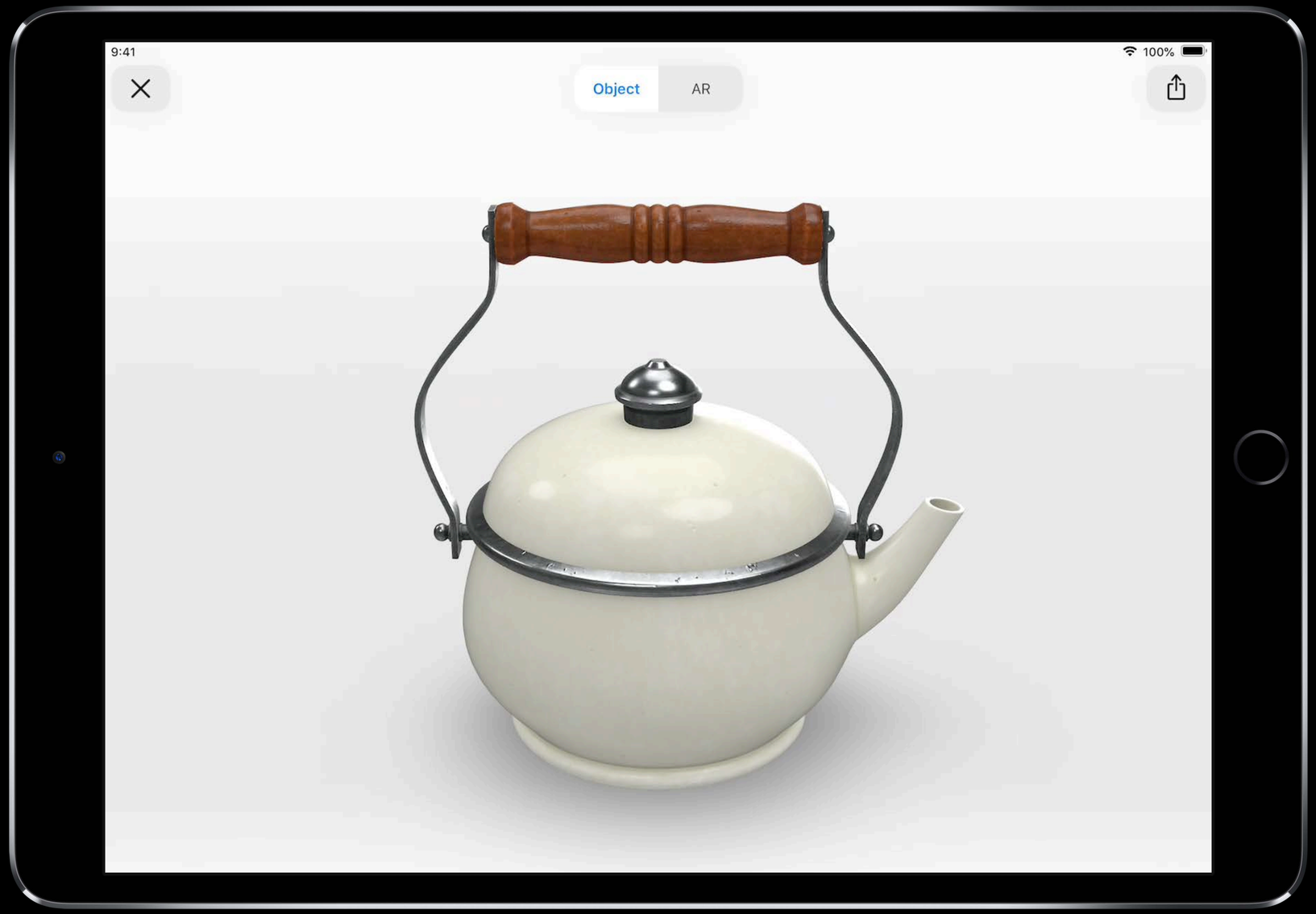
Placement



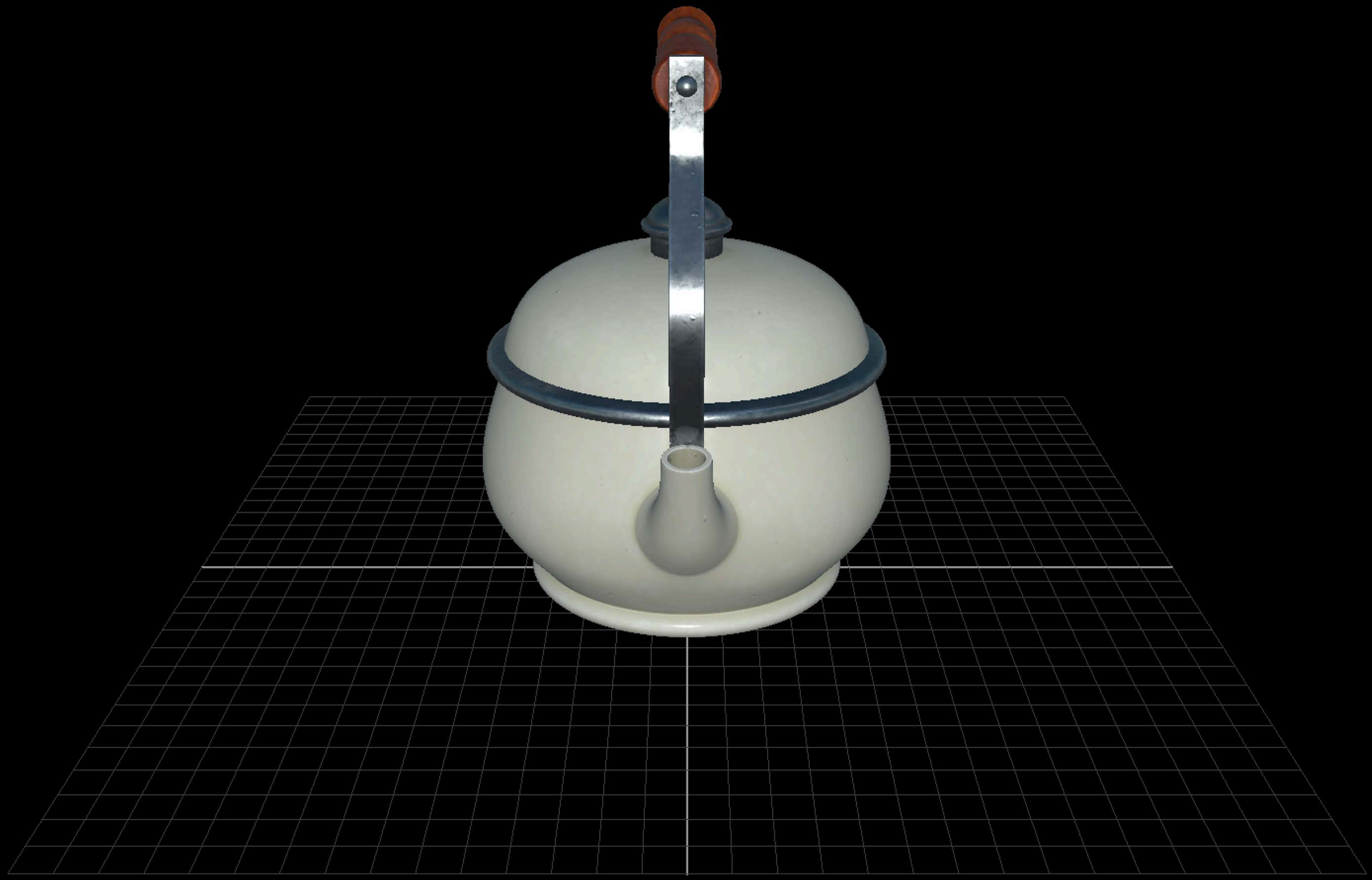
Placement



Placement

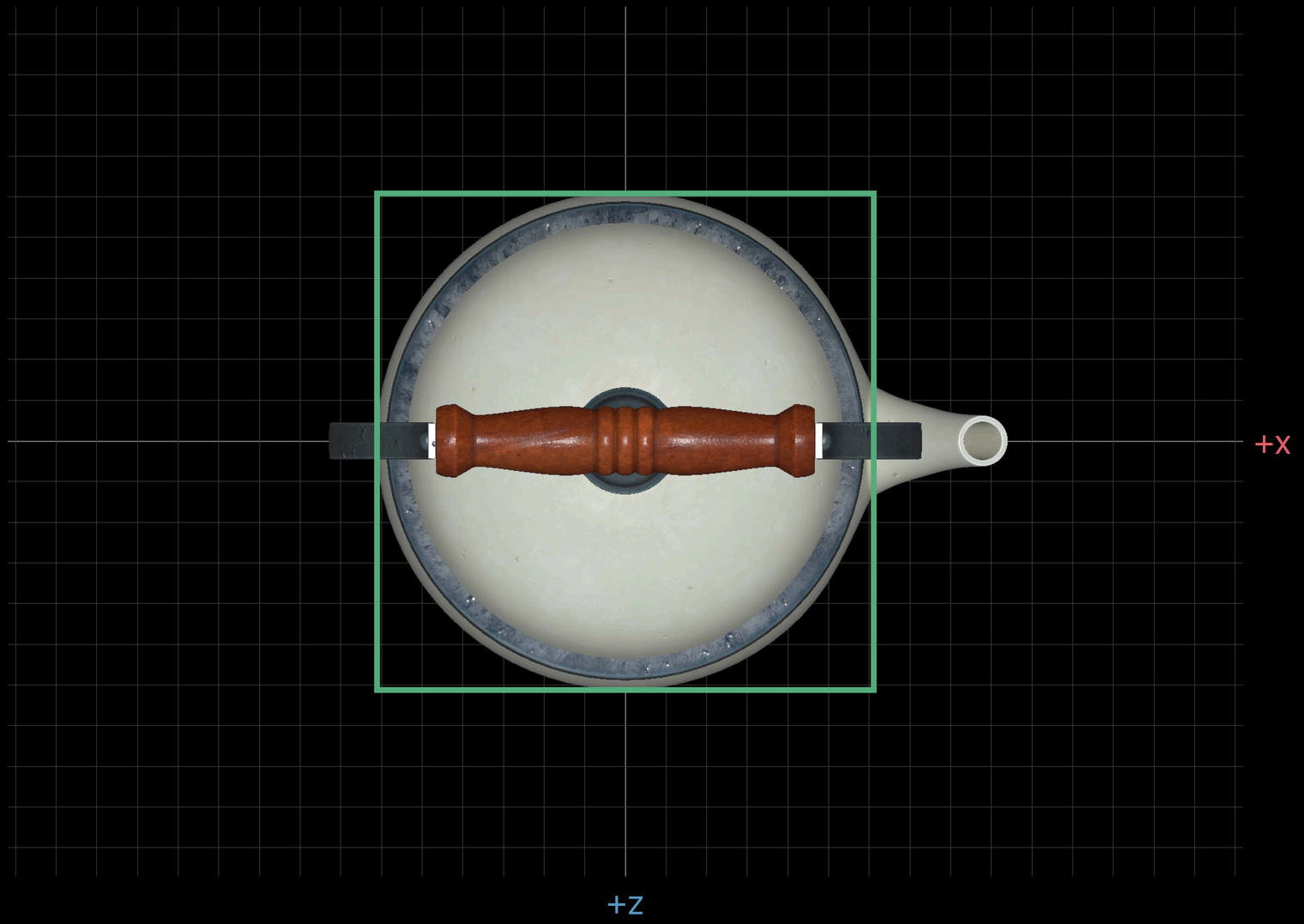

9:41
100%
Object
AR

Placement

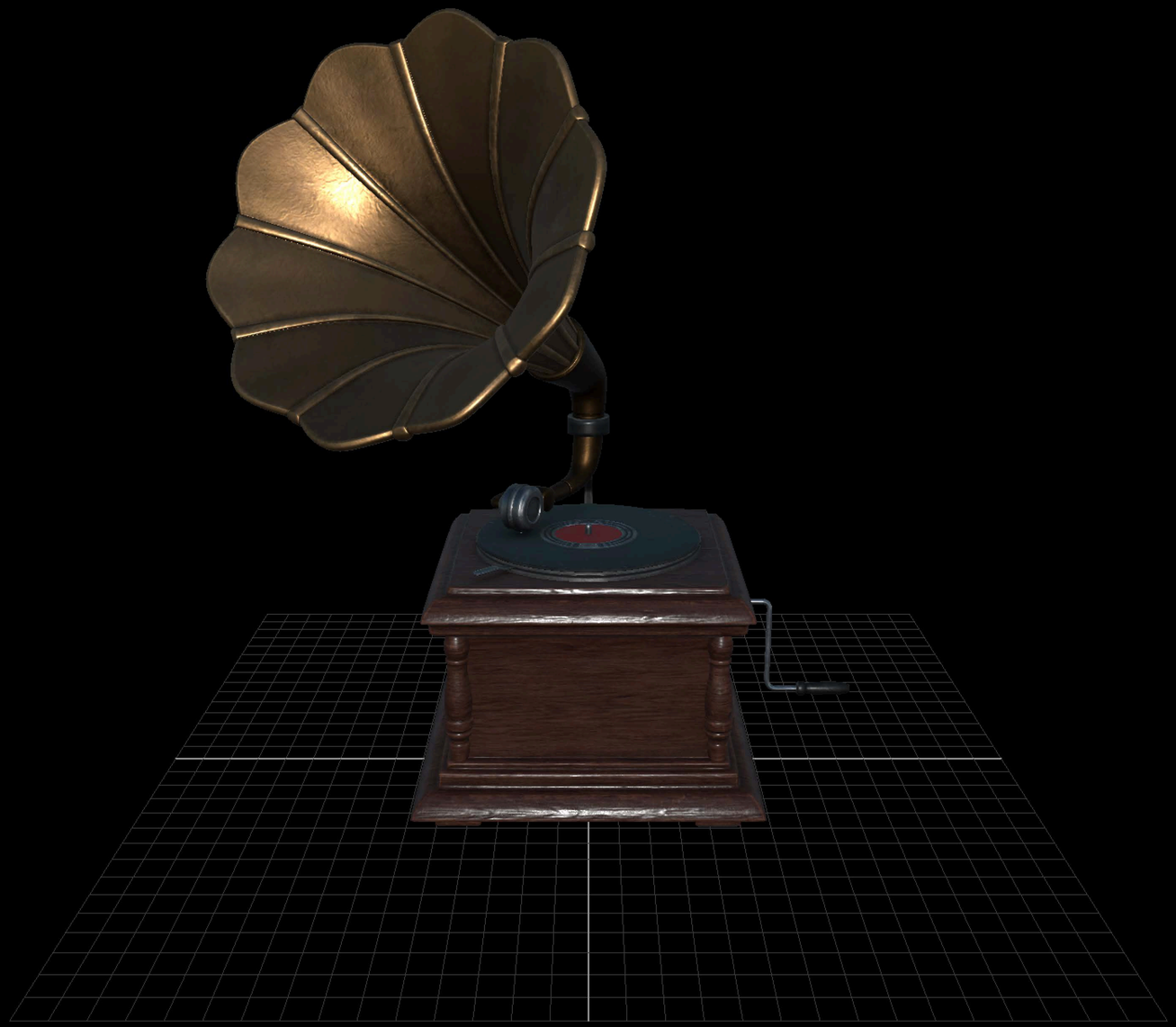


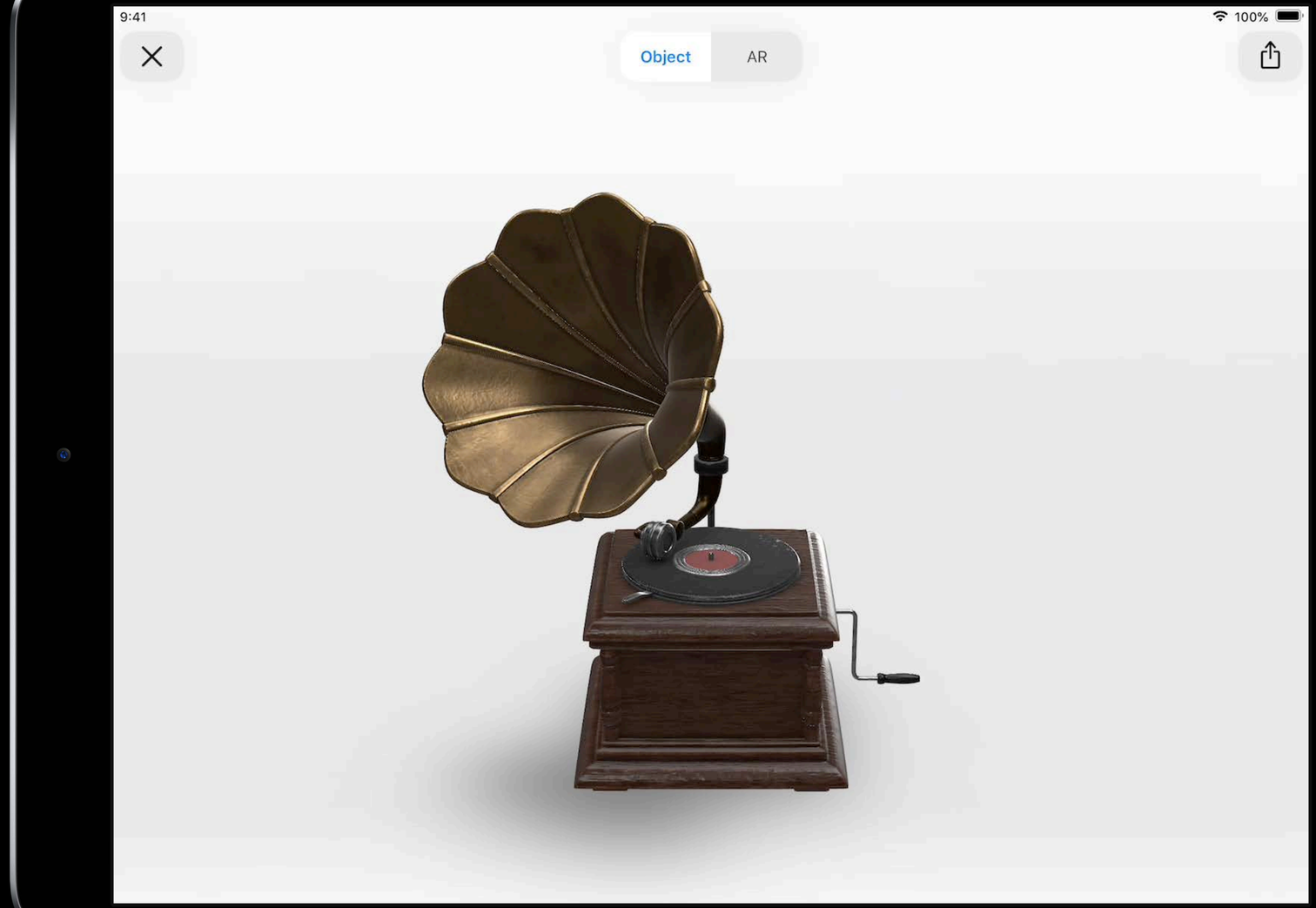
Placement

Placement

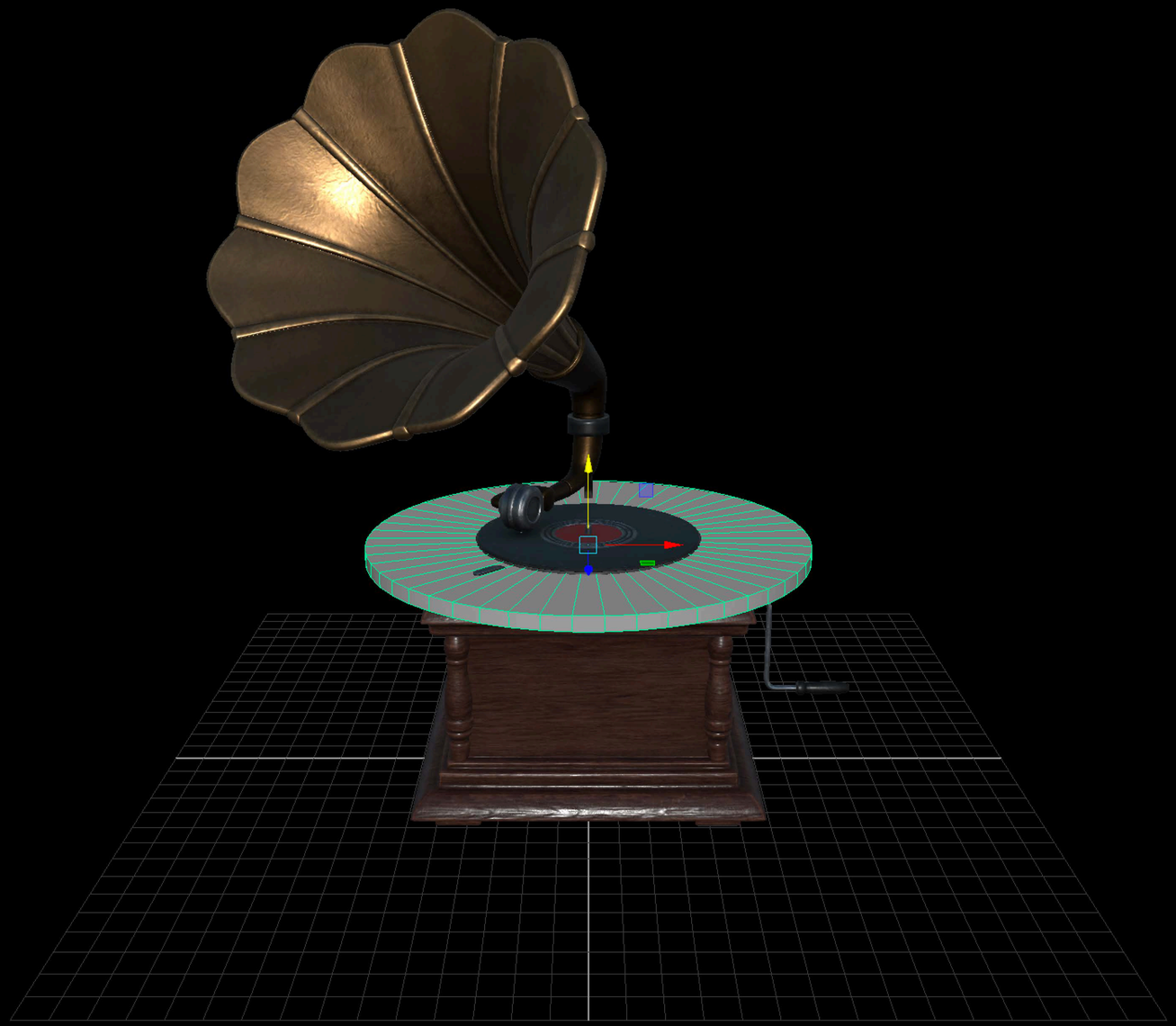


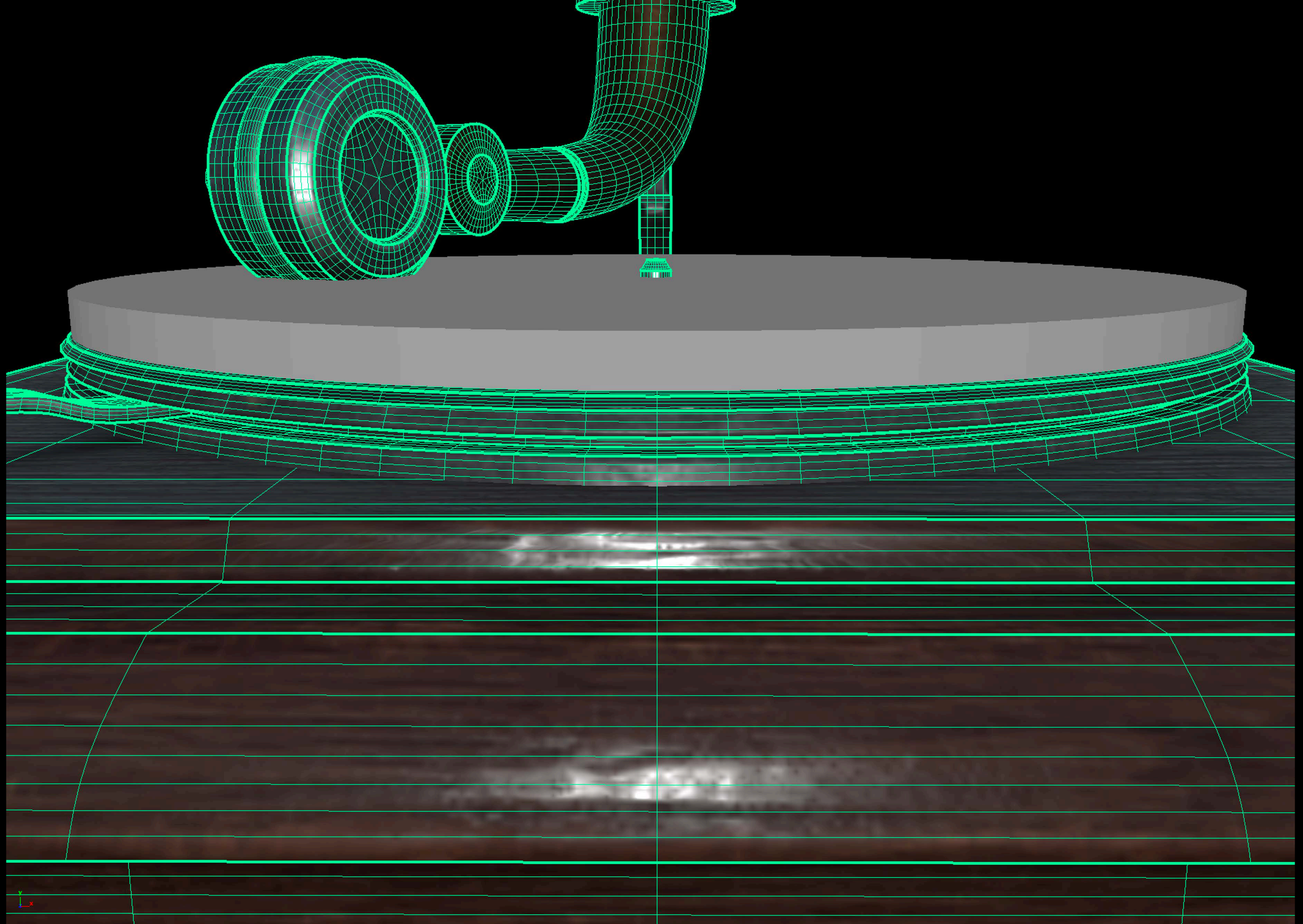
Physical Size



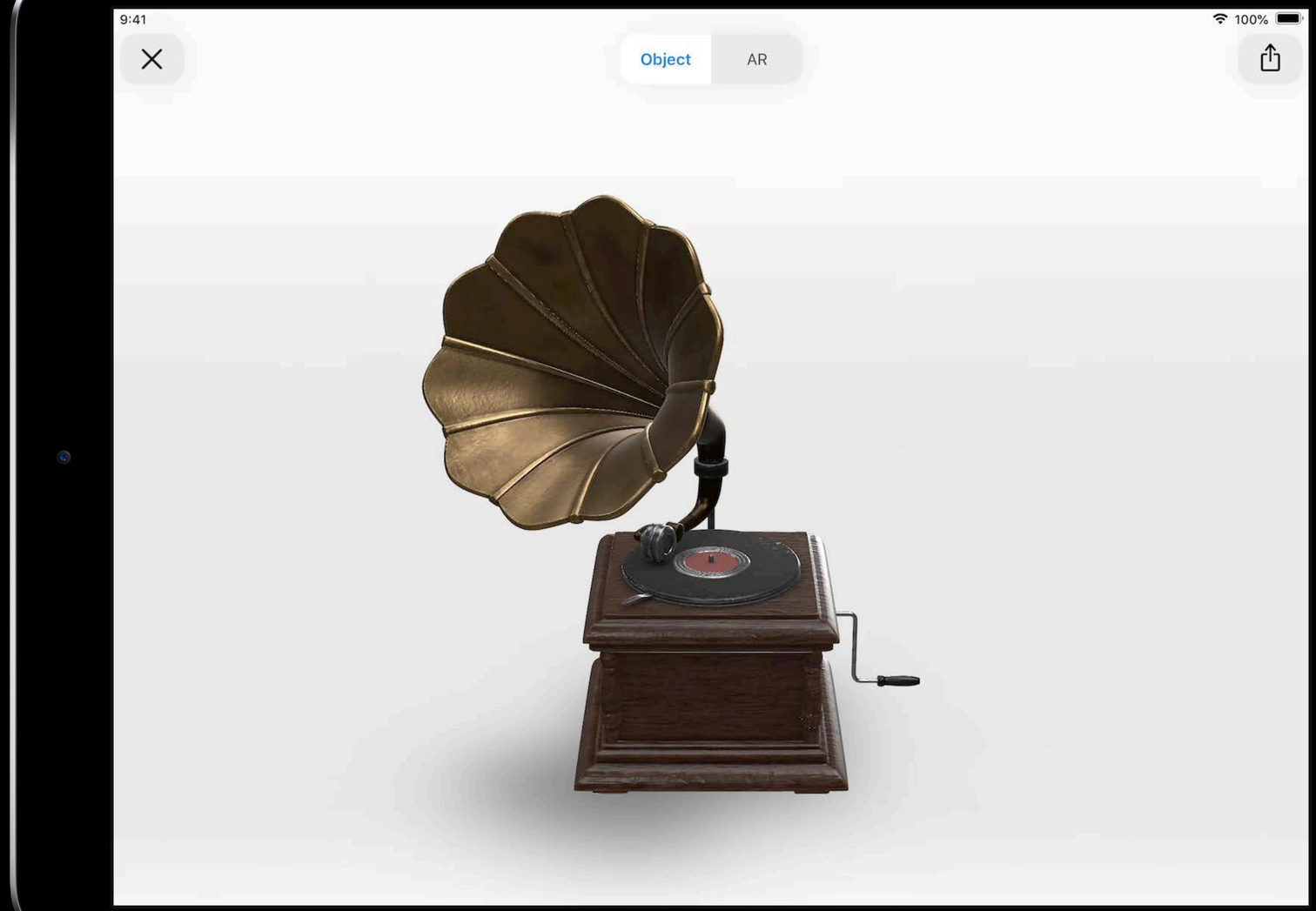

9:41
100%
Object
AR

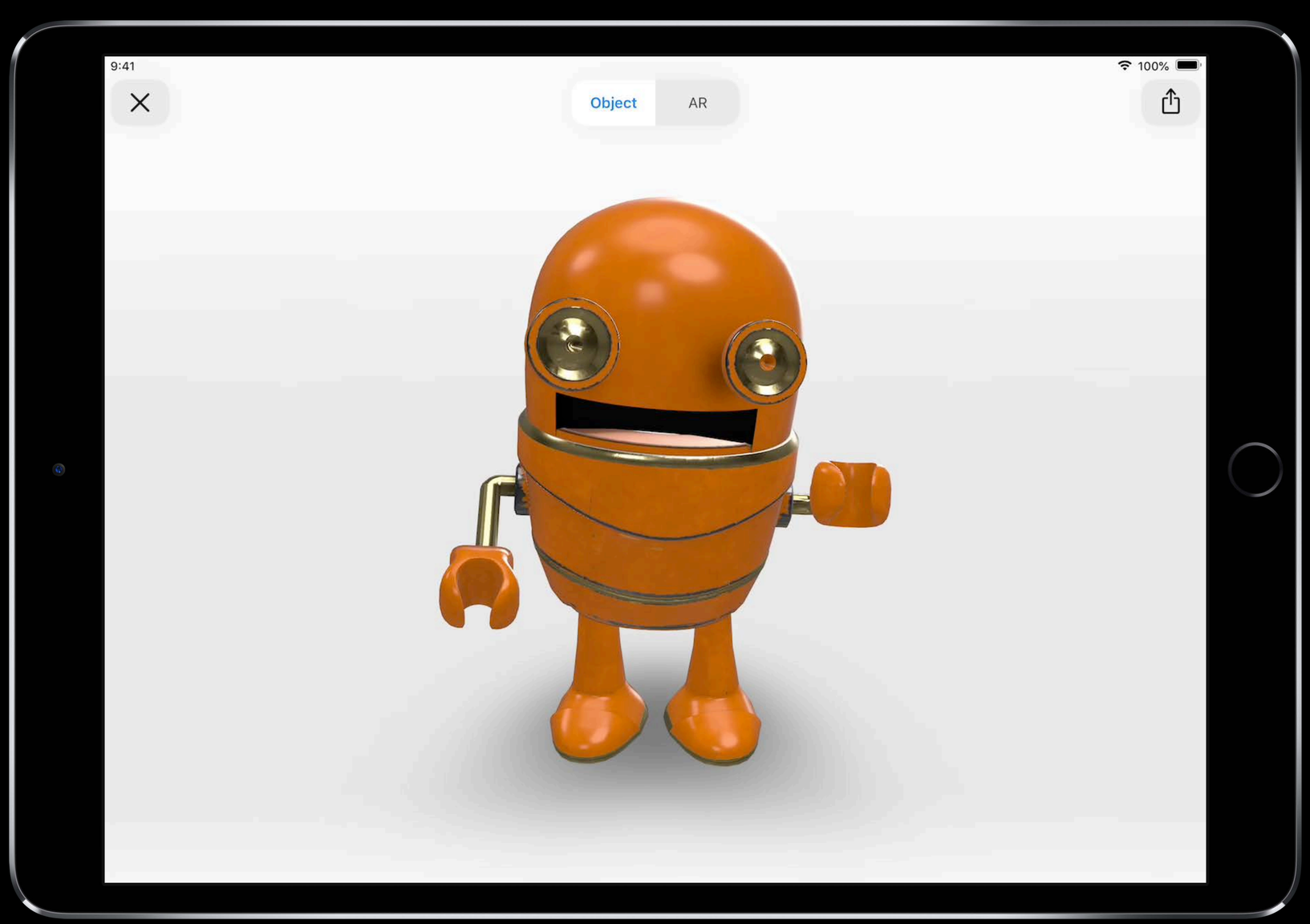
Physical Size





Physical Size


9:41
100%
Object
AR

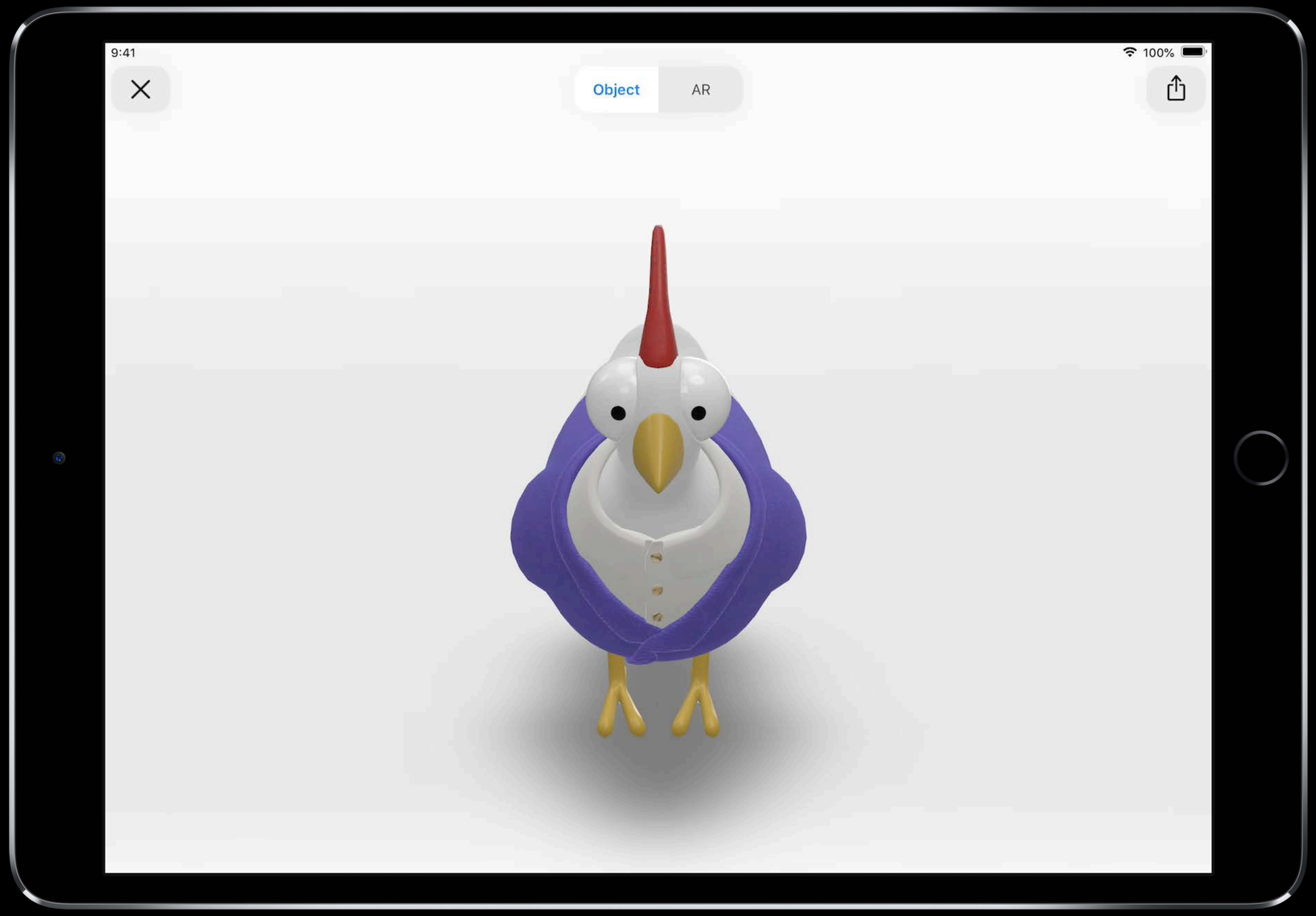

9:41
100%
Object
AR

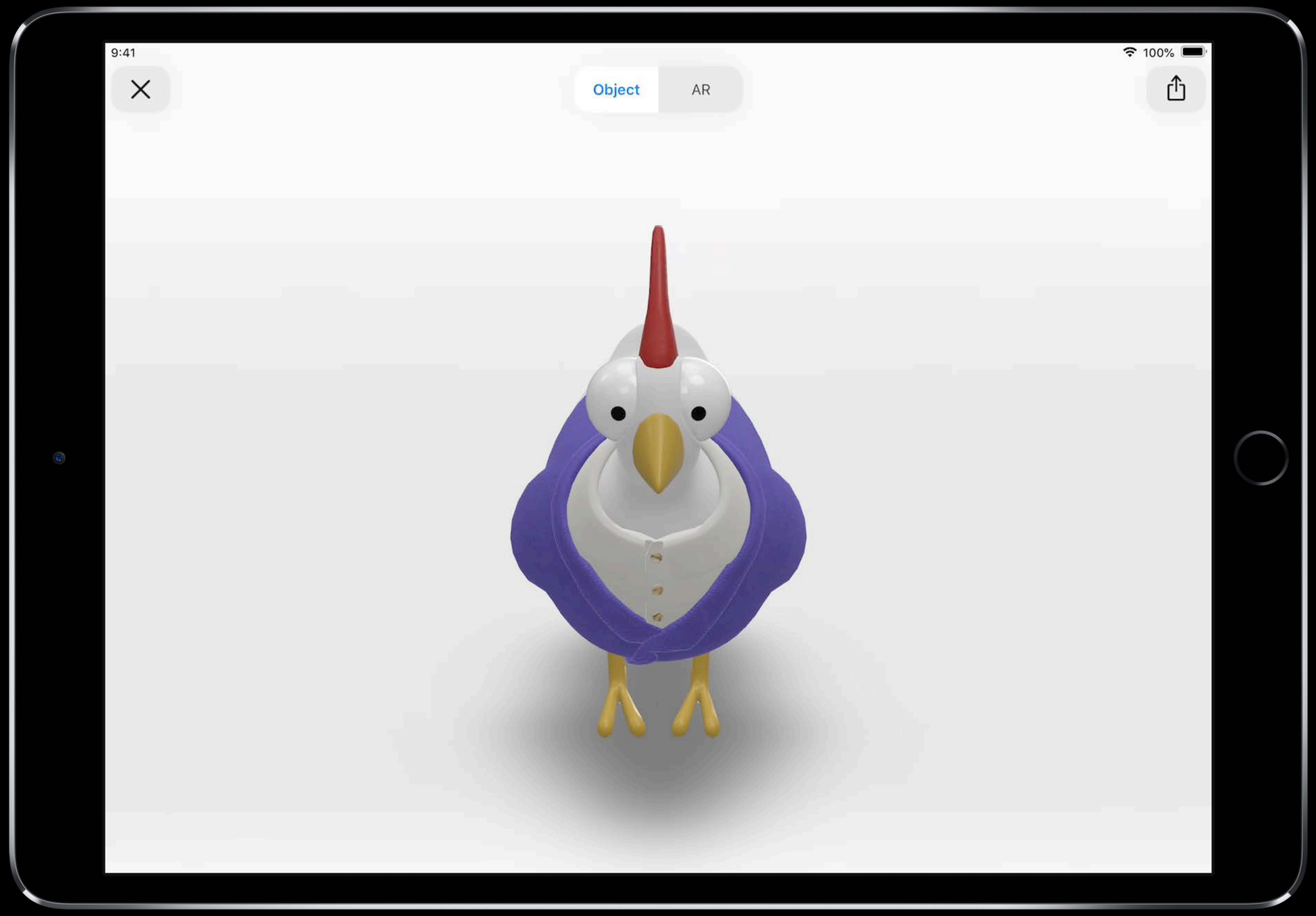
Animation

Provide an “idle” animation to add life to the object

Animations always loop

Animations can be a mix of skeletal animation and transform animation


9:41
100%
Object
AR


9:41
100%
Object
AR

Animation Tips

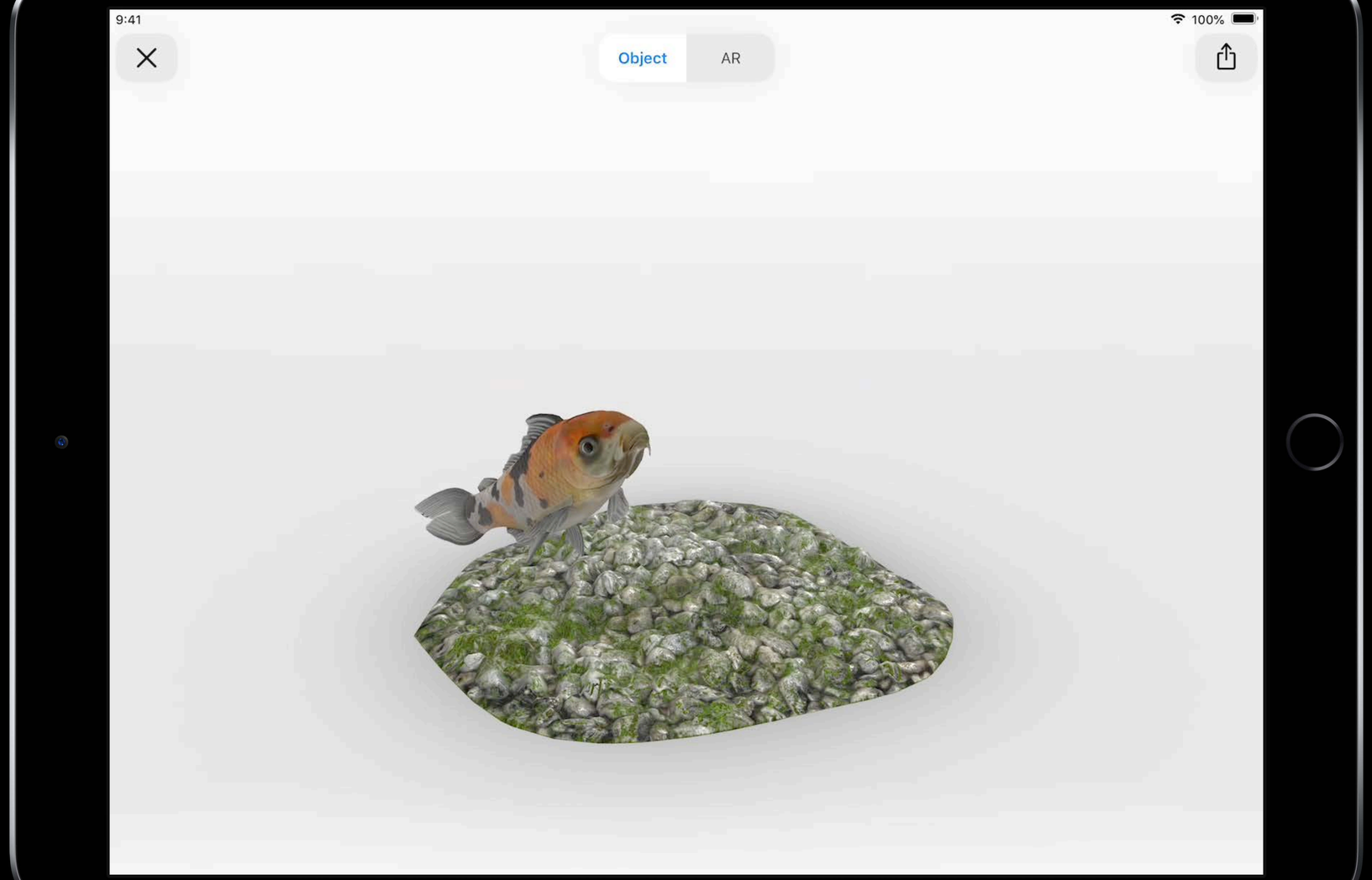
Choose animations that enhance AR immersiveness

Don’t animate objects away from the origin

Keep a consistent bounding box throughout an animation

Prefer animations that make sense at a static location

Or create animations as self-contained scenes


9:41
100%
Object
AR

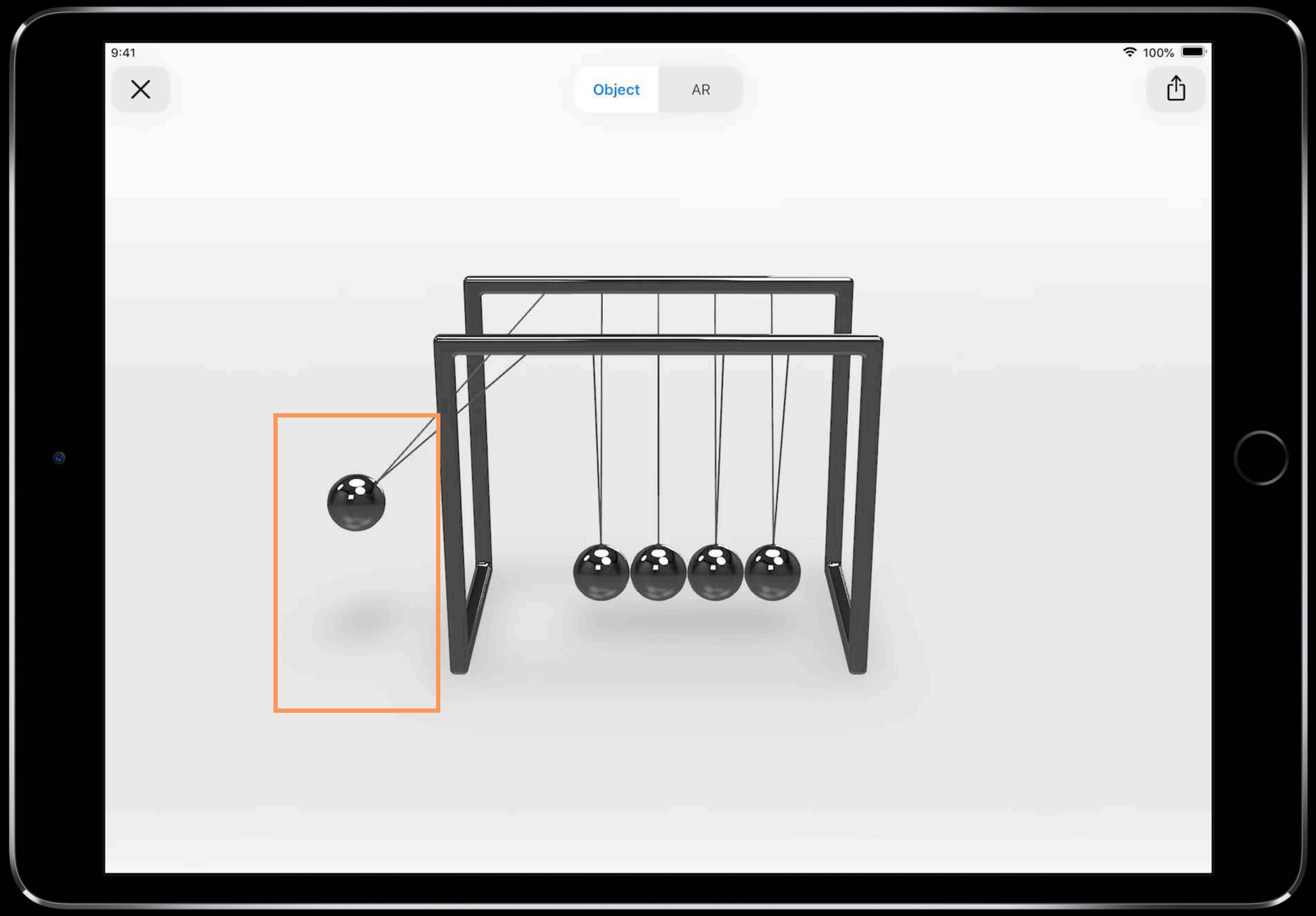
Contact Shadow

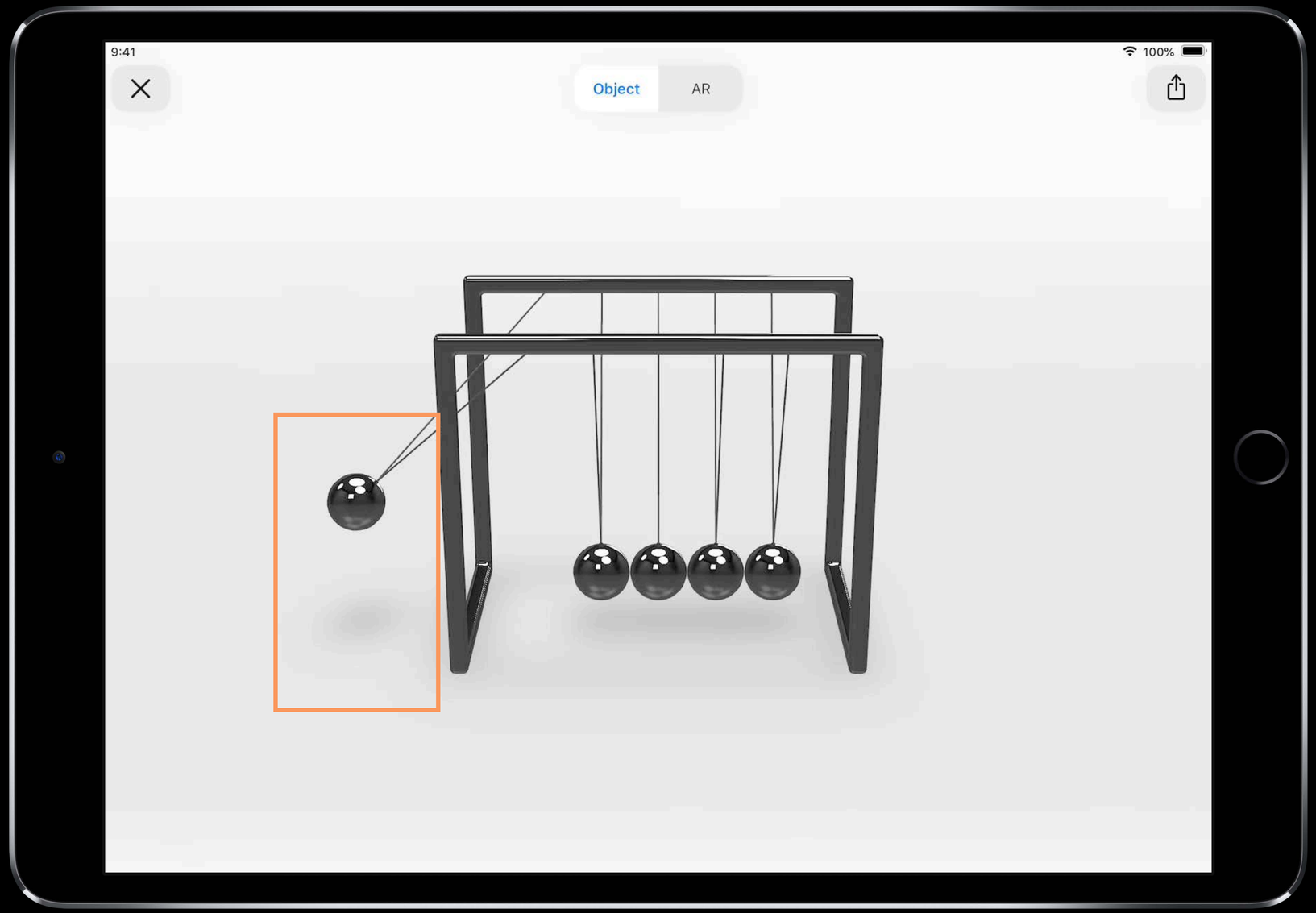
AR Quick Look provides a contact shadow for you

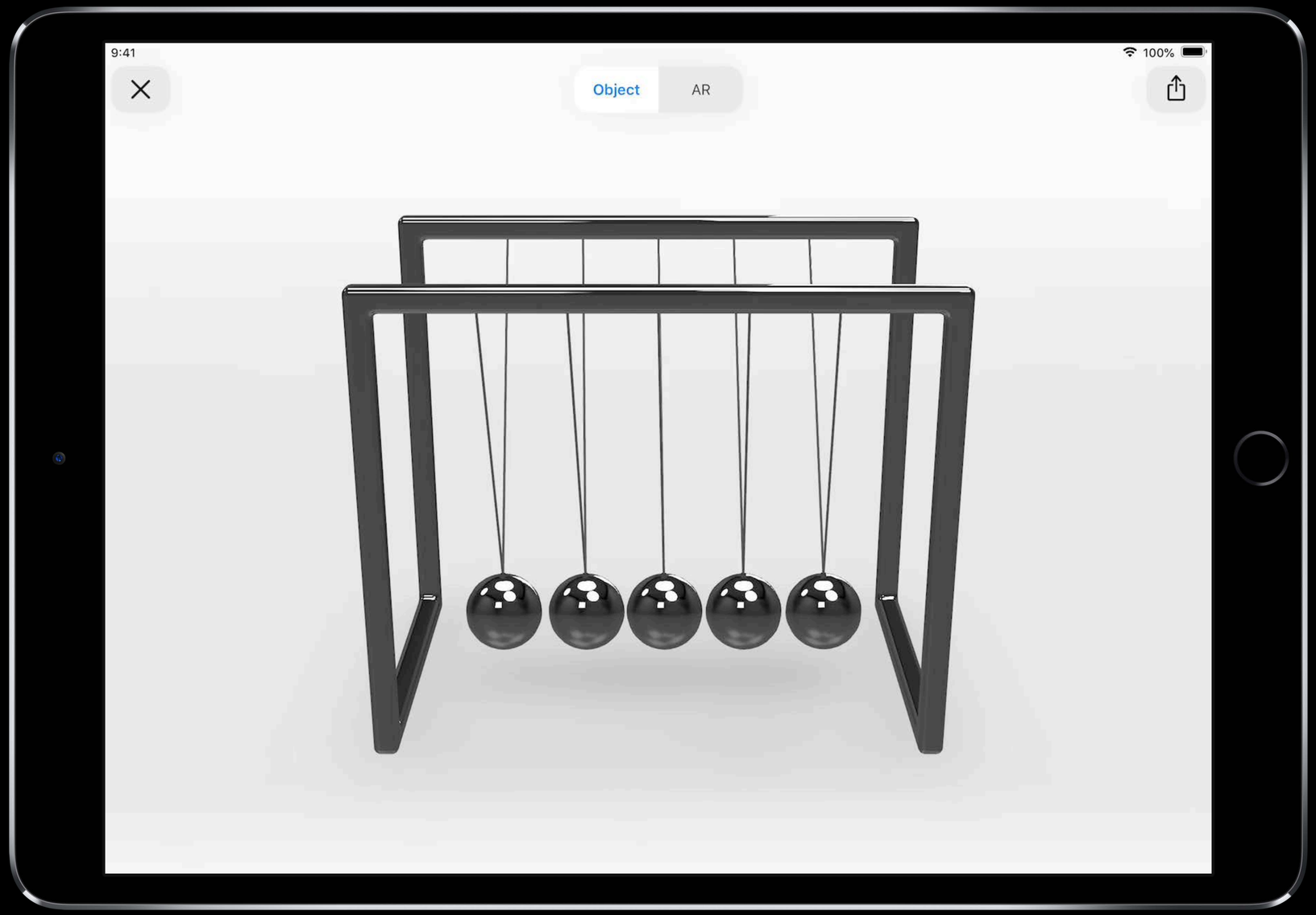
- Can turn the shadow on and off
- Can apply ambient lighting conditions

Don’t bake a contact shadow into the model

First animation frame is used for shadow creation


9:41
100%
Object
AR


9:41
100%
Object
AR

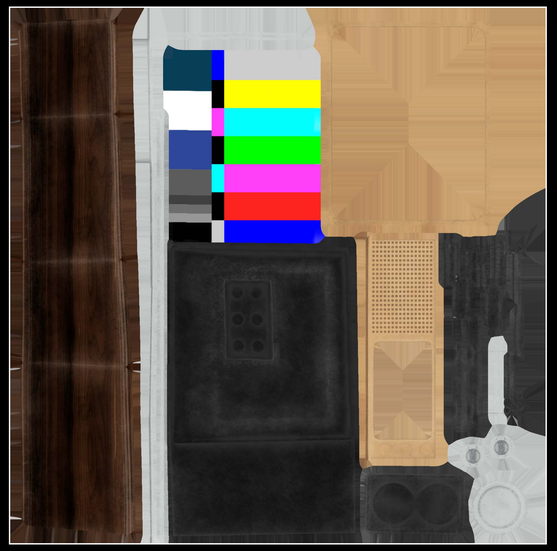

9:41
100%
Object
AR

Model Appearance

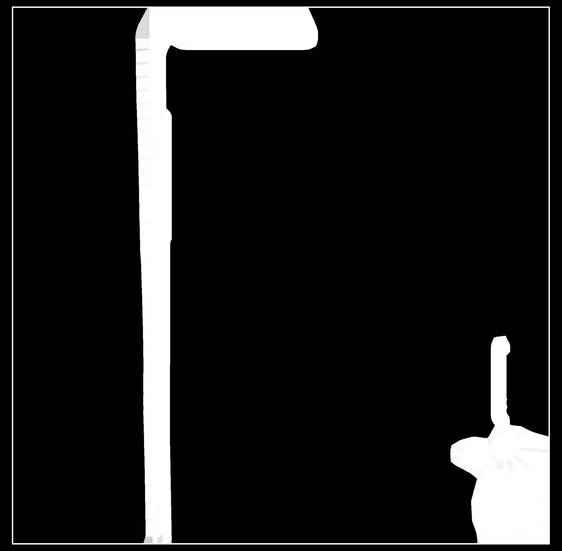
AR Quick Look uses a Physically Based Rendering (PBR) shader

- Albedo (base color)
- Metallic (conductor or insulator)
- Roughness (rough or shiny)
- Normal (surface details)
- Ambient occlusion (internal shadows)
- Emissive (emits light)

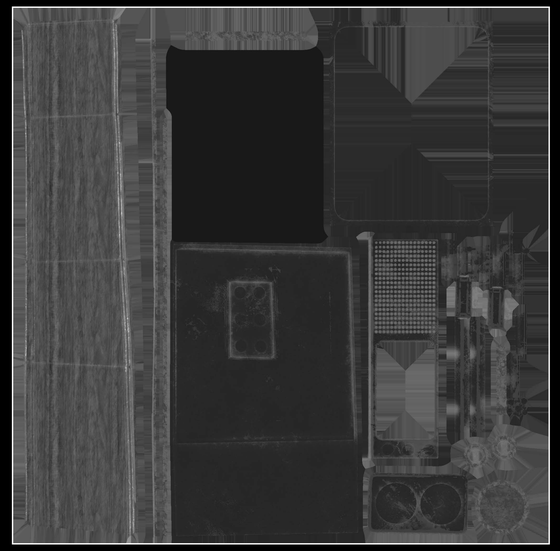
No textures



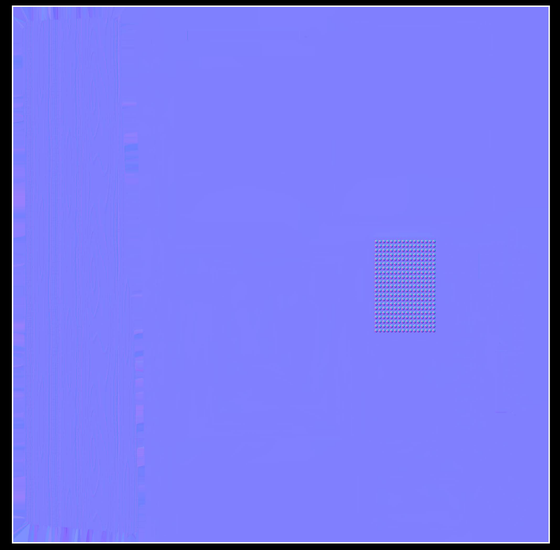
Albedo



Metallic

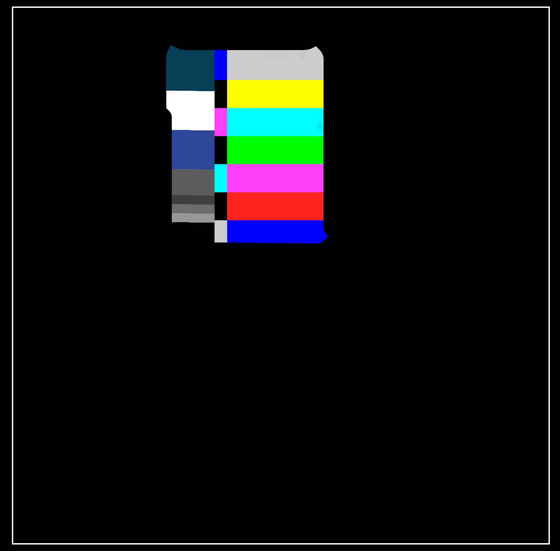


Roughness



Normal

Ambient Occlusion



Emissive

Transparency

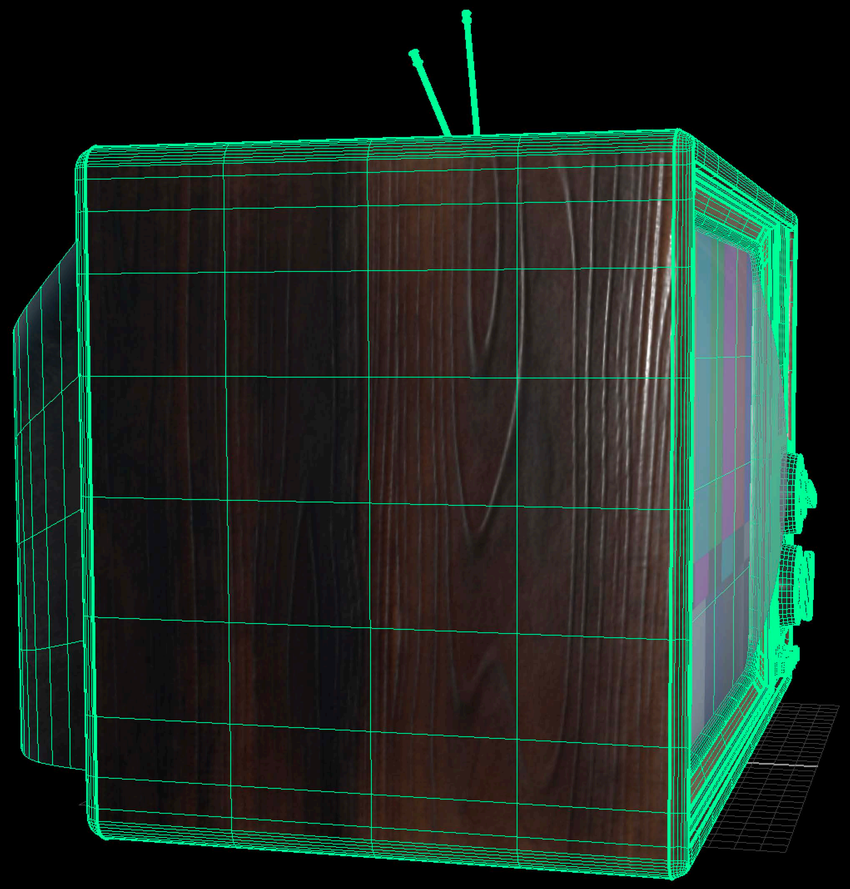
Use a separate material for transparent and non-transparent parts of the model

Provide an albedo texture with transparency in its alpha channel

Use transparency for see-through areas, not for cutouts

Transparency

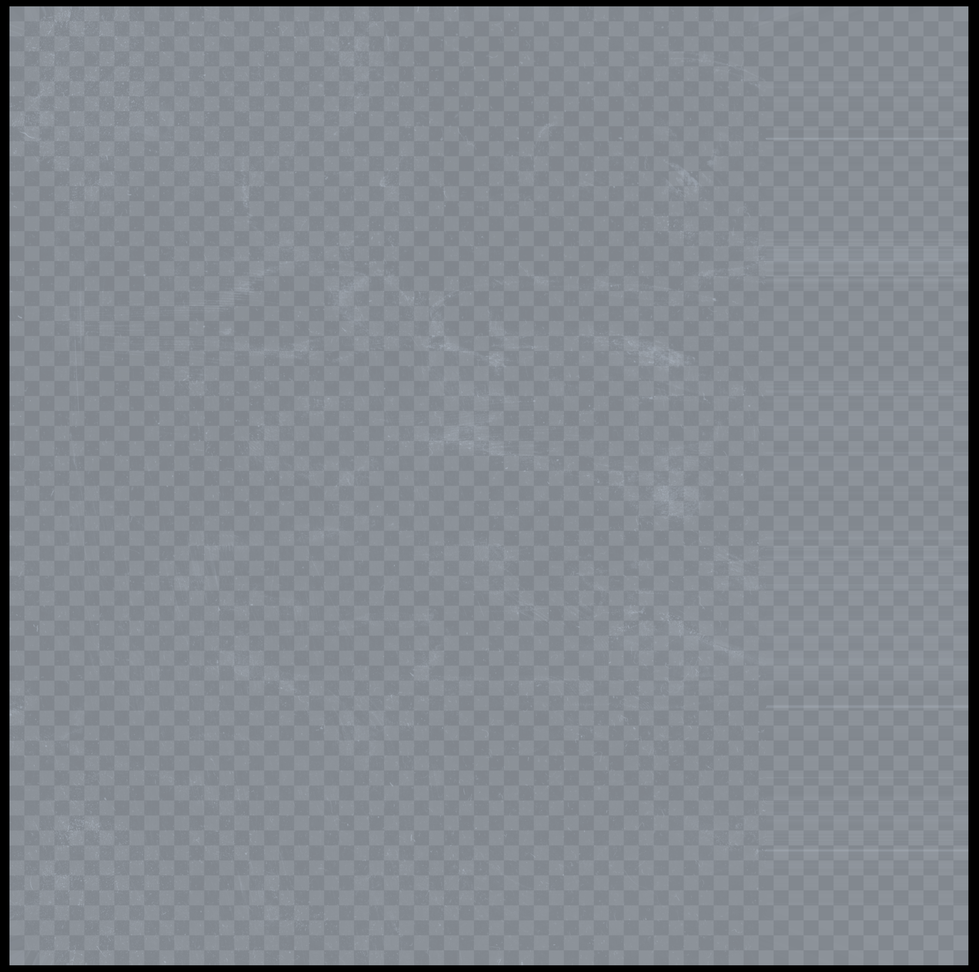
Transparency



Transparency


9:41
100%
Object
AR

Transparency



Texture Formats

- Albedo (RGB/RGBA)
- Metallic (Grayscale)
- Roughness (Grayscale)
- Normal (RGB)
- Ambient occlusion (Grayscale)
- Emissive (RGB)

Any image format supported by iOS

Textures should be square powers of 2 (2048, 1024, 512…)

Supporting All Devices

System-wide extension that shares system memory with applications

Optimize and test your models for high-memory devices

- iPhone 7 Plus, iPhone 8 Plus, iPhone X
- iPad Pro 12.9”

AR Quick Look will dynamically downsample textures for other devices when needed

Meshes and animations are not modified

Size Limits

Many factors affect a model’s memory requirements

- Mesh and animation complexity
- Texture size and count

As a guide, for a model with a single PBR material

- 100k polygons
- One set of 2048 by 2048 PBR textures
- Ten seconds of animation

Always test your model on a real device

Optimizing Models

Freeze transforms and merge adjacent vertices

If possible, use a single material/texture set for the entire model

Don’t include textures you don’t need

Spend your texture budget on areas that add most value and realism

Remember that pixels have a physical size in AR

Balance texture size and quality against download size

‘usdz’ Converter

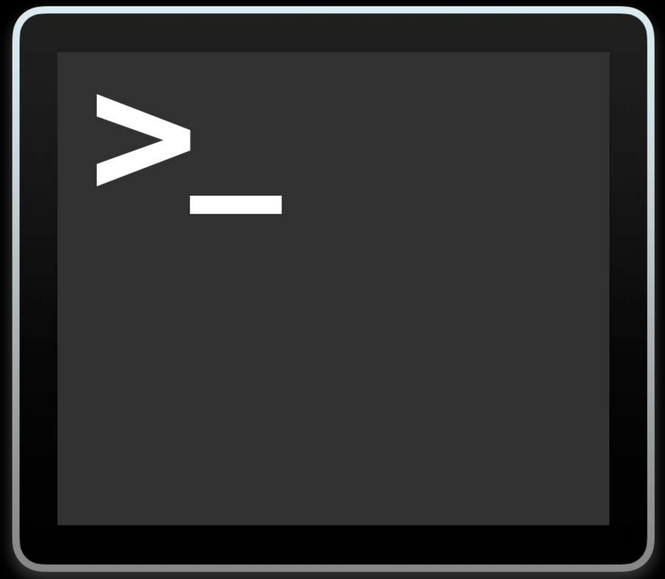
Command line tool to convert 3D models
to .usdz format

Ships with Xcode 10

Maps PBR textures to meshes and submeshes

Input formats

- OBJ file
- Single-frame Alembic (ABC) file
- USD file (either .usda or .usdc)



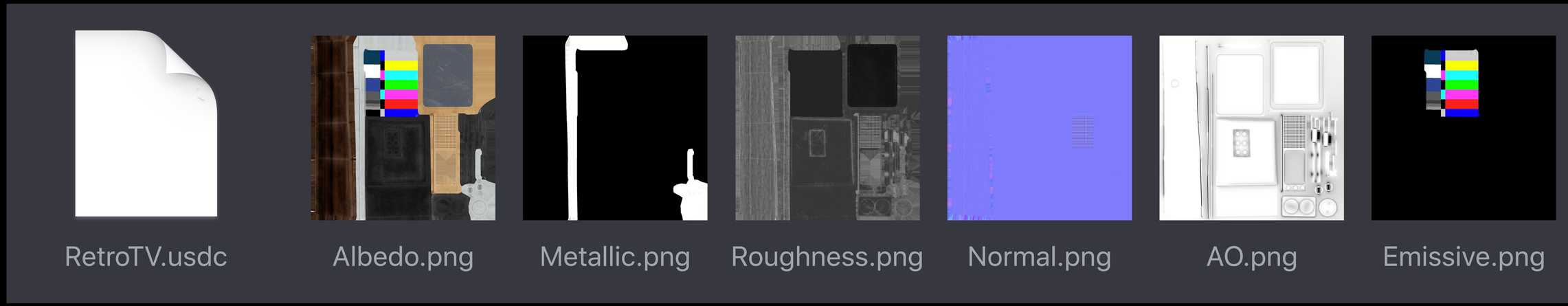
Anatomy of a ‘usdz’ File

A .usdz file is an uncompressed zip archive

First file in the archive is a .usdc file (model, animation, material definitions)

Remaining files (if needed) are image files

RetroTV.usdz



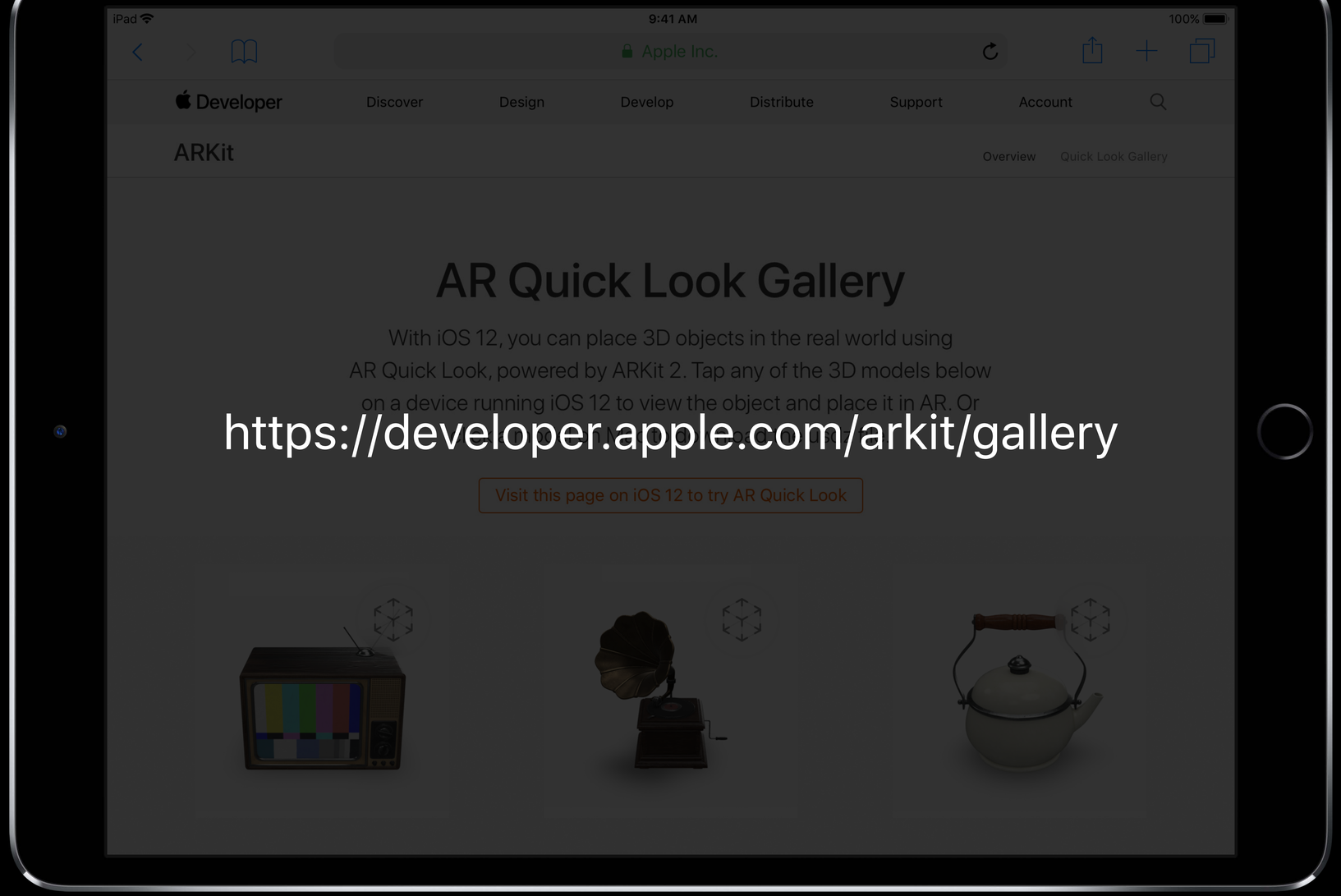
https://graphics.pixar.com/usd/

```
// Call usdz_converter with xcrun.
xcrun usdz_converter RetroTV.obj RetroTV.usdz
```

```
// Call usdz_converter with xcrun.
xcrun usdz_converter RetroTV.obj RetroTV.usdz
```

```
// PBR textures can be applied to groups (meshes and submeshes) with the -g option.
xcrun usdz_converter RetroTV.obj RetroTV.usdz
    -g RetroTVMesh
    -color_map RetroTV_Albedo.png
    -metallic_map RetroTV_Metallic.png
    -roughness_map RetroTV_Roughness.png
    -normal_map RetroTV_Normal.png
    -ao_map RetroTV_AmbientOcclusion.png
    -emissive_map RetroTV_Emissive.png
```

```
// Call usdz_converter with xcrun.
xcrun usdz_converter RetroTV.obj RetroTV.usdz

// PBR textures can be applied to groups (meshes and submeshes) with the -g option.
xcrun usdz_converter RetroTV.obj RetroTV.usdz
    -g RetroTVMesh
    -color_map RetroTV_Albedo.png
    -metallic_map RetroTV_Metallic.png
    -roughness_map RetroTV_Roughness.png
    -normal_map RetroTV_Normal.png
    -ao_map RetroTV_AmbientOcclusion.png
    -emissive_map RetroTV_Emissive.png

// Use the -v option to print out group names and other verbose information during conversion.
xcrun usdz_converter RetroTV.obj RetroTV.usdz -v
```


iPad
9:41 AM
100%
Apple Inc.
Developer
Discover
Design
Develop
Distribute
Support
Account
ARKit
Overview
Quick Look Gallery
AR Quick Look Gallery
With iOS 12, you can place 3D objects in the real world using
AR Quick Look, powered by ARKit 2. Tap any of the 3D models below
on a device running iOS 12 to view the object and place it in AR. Or
https://developer.apple.com/arkit/gallery
Visit this page on iOS 12 to try AR Quick Look

Summary

A system-wide way to view AR content in the real world

Can be integrated into your own apps and websites

Uses the .usdz file format for 3D model distribution and sharing

Supports PBR, animation, and transparency

`usdz_converter` tool in Xcode 10 to convert existing models to ‘usdz’

More Information

https://developer.apple.com/wwdc18/603

ARKit Lab	Technology Lab 5	Tuesday 12:00PM
ARKit Lab	Technology Lab 5	Wednesday 3:00PM
ARKit Lab	Technology Lab 5	Friday 9:00AM


WWDC18