



Magic Markers

Max McCoy

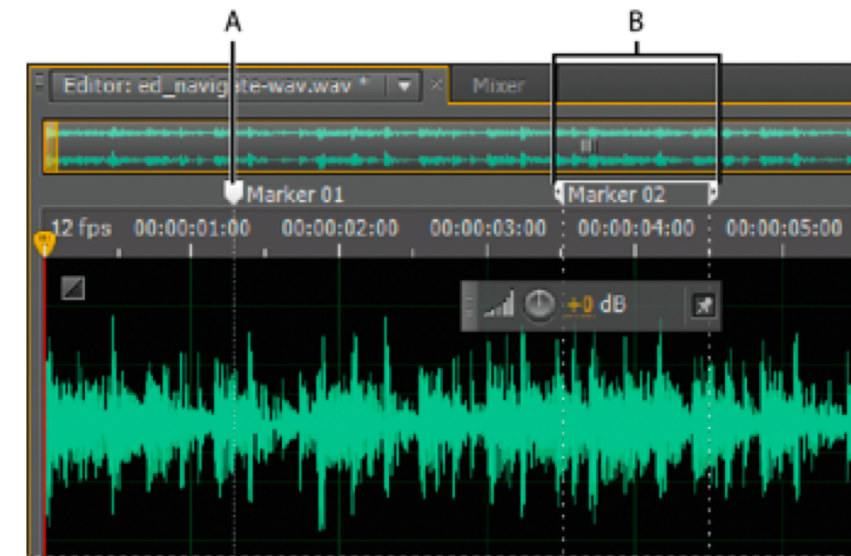
Technical Sound Designer – Disney Games and Interactive Experiences

GAME DEVELOPERS CONFERENCE

MARCH 18–22, 2019 | #GDC19

Marker Overview

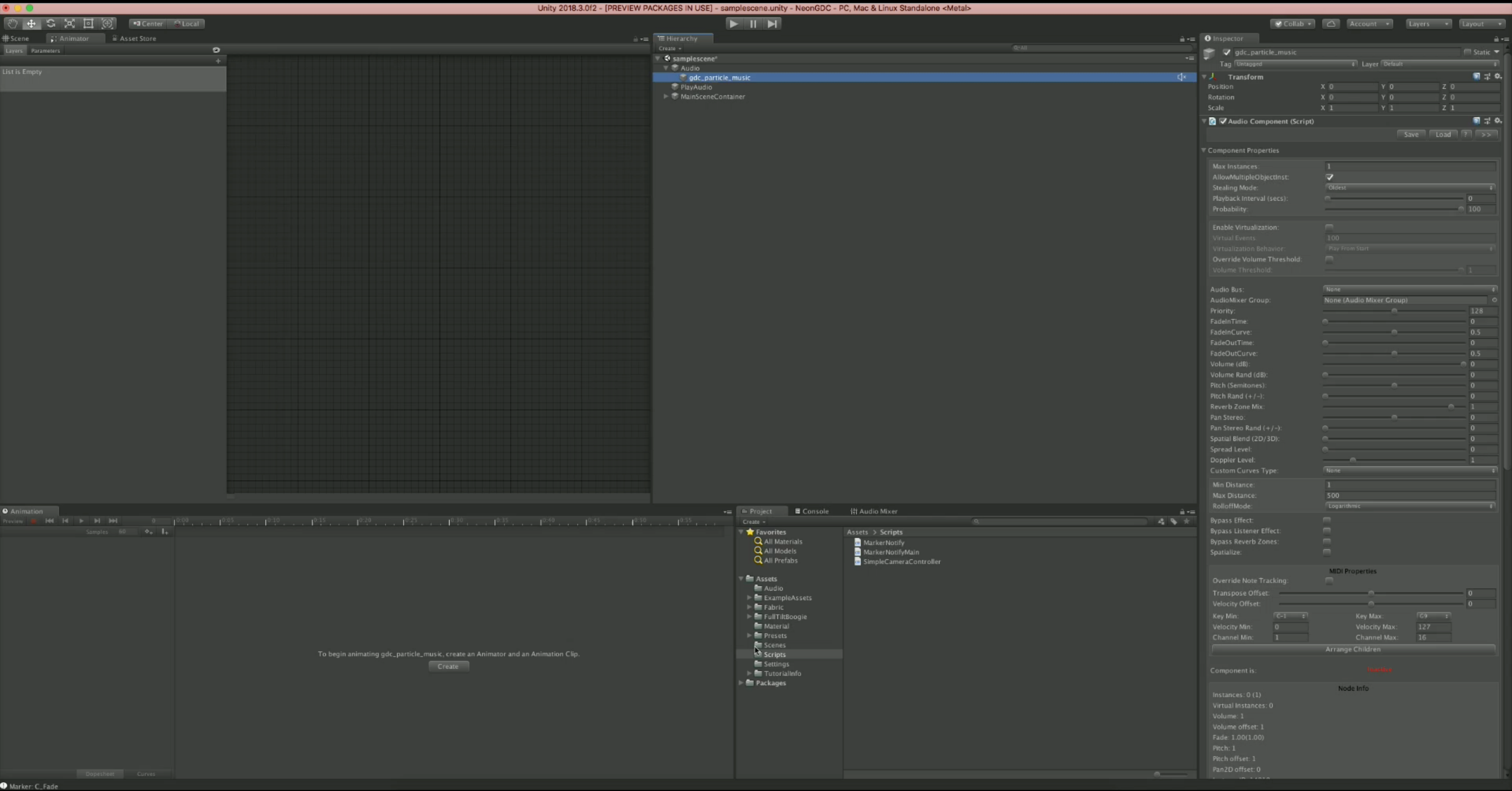
- What are markers?
 - Markers are identifiers that are inserted in a wav file and used to tag position in the waveform.
 - Markers are usually created in a wav editor.
 - A few examples are Soundforge, Adobe Audition, or Cue Tool.
 - Logic



Particles

Opened in 0.98 seconds

40000 Hz • 24-bit • Stereo	0.50 MB	0.50.747	400.20 GB free
----------------------------	---------	----------	----------------



```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class MarkerNotifyMain : MonoBehaviour
6  {
7      //Event for Notify
8      // Fabric.EventManager.Instance.PostEventNotify("EventName", gameObject, Notify);
9
10     /// <summary>
11     /// Notify the specified type, eventName, info and gameObject.
12     /// </summary>
13     /// <param name="type">Type.</param>
14     /// <param name="eventName">Event name.</param>
15     /// <param name="info">Info.</param>
16     /// <param name="gameObject">Game object.</param>
17
18     public void Notify(Fabric.EventNotificationType type, string eventName, object info, GameObject gameObject)
19     {
20         if (info != null)
21         {
22             if (type == Fabric.EventNotificationType.OnMarker)
23             {
24                 Fabric.MarkerNotificationData c = info as Fabric.MarkerNotificationData;
25                 if (c != null)
26                 {
27                     //This is where you detect the marker and do an action
28                     Debug.Log("Marker: " + c._label);
29                 }
30             }
31         }
32     }
33 }
34
35

```

```

1 using System.Collections;
2 using System.Collections.Generic;
3 Switch to next document e;
4
5 public class MarkerNotify : MonoBehaviour {
6
7     public GameObject gObject;
8     public GameObject dObject;
9     public GameObject cObject;
10    public ParticleSystem particleOne;
11    public ParticleSystem particleTwo;
12
13
14    // Event for Notify ...
15
16    private void Start()
17    {
18        Invoke("PlayMusic", 5f);
19    }
20
21    /// <summary> Notify the specified type, eventName, info and gameObject.
22
23
24    public void Notify(Fabric.EventNotificationType type, string eventName, object info, GameObject gameObject)
25    {
26        if (info != null)
27        {
28            if (type == Fabric.EventNotificationType.OnMarker)
29            {
30                Fabric.MarkerNotificationData c = info as Fabric.MarkerNotificationData;
31                if (c != null)
32                {
33                    Debug.Log("Marker: " + c._label);
34
35                    if(c._label == "G_Fade")
36                    {
37                        gObject.SetActive(true);
38                        dObject.SetActive(false);
39                        cObject.SetActive(false);
40                    }
41                    else if(c._label == "D_Fade")
42                    {
43                        gObject.SetActive(false);
44                        dObject.SetActive(true);
45                        cObject.SetActive(false);
46                    }
47                    else if (c._label == "C_Fade")
48                    {
49                        gObject.SetActive(false);
50                        dObject.SetActive(false);
51                        cObject.SetActive(true);
52                    }
53                    else if(c._label == "On")
54                    {
55                        gObject.SetActive(true);
56                        dObject.SetActive(true);
57                        cObject.SetActive(true);
58                    }
59                    else if (c._label == "Off")
60                    {
61                        gObject.SetActive(false);
62                        dObject.SetActive(false);
63                        cObject.SetActive(false);
64                    }
65                    else if(c._label == "PlayParticle")
66                    {
67                        particleOne.Play();
68                    }
69                    else if(c._label == "PlayParticle2")
70                    {
71                        particleTwo.Play();
72                    }
73                }
74            }
75        }
76    }
77
78    }
79
80    }
81
82    }
83
84    public void PlayMusic()
85    {
86        Fabric.EventManager.Instance.PostEventNotify("music", gameObject, Notify);
87    }
88
89
90 }
91

```


Animations

Animations

- Markers can be used to trigger animations
- Dialog files can be used for lip flap, or gestures
- Music files can inform cinematic moments

Waveform

Multitrack

Files

Favorites

📁

📄

🔍

🗑️

Name ↑

Status

Duration

Sample Rate

Channels

Bit Depth

Source Fo

>

++

GDC_RobotAnim_1wav

0:27.930

48000 Hz

Stereo

24

Waveform

▶

📄

🔊

Media Browser

Effects Rack

Markers

Properties

📁

🗑️

🔍

📄

🔍

🔊

Name	Start ↓	End	Duration	Type	Description
win_A	0:25.312		0:00.000	Cue	▼
hp_straight_right_A	0:25.009		0:00.000	Cue	▼
hp_straight_A	0:24.576		0:00.000	Cue	▼
hp_upper_right_A	0:23.216		0:00.000	Cue	▼
ho_hook_left_A	0:22.810		0:00.000	Cue	▼
hp_straight_right_A	0:22.377		0:00.000	Cue	▼
hp_upper_right_A	0:21.029		0:00.000	Cue	▼
ho_hook_left_A	0:20.633		0:00.000	Cue	▼
hp_straight_right_A	0:20.194		0:00.000	Cue	▼
hp_straight_A	0:19.091		0:00.000	Cue	▼
Ride_The_Bull_01	0:14.689		0:00.000	Cue	▼
Robo-Taunt	0:05.514		0:00.000	Cue	▼
Bow	0:00.205		0:00.000	Cue	▼

History

Video

▶

📄

Open

fx Amplify

0 Undo

🗑️

Apply Amplify completed in 0.04 seconds

Editor: GDC_RobotAnim_1wav

Mixer

🔊

🔊

🔊

🔊

🔊

🔊

🔊

🔊

🔊

🔊

hms

2.0

4.0

6.0

8.0

10.0

12.0

14.0

16.0

18.0

20.0

22.0

24.0

26.0

2

🔊

🔊

🔊

🔊

🔊

🔊

🔊

🔊

🔊

🔊

dB

-1

-3

-6

-9

-12

-18

-24

-∞

-24

-18

-12

-6

-3

-1

L

dB

-1

-3

-6

-9

-12

-18

-24

-∞

-24

-18

-12

-6

-3

-1

R

0:00.000

▶

⏸

⏮

⏪

⏩

⏭

🔴

📄

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

Levels

🔍

dB

-59

-58

-57

-56

-55

-54

-53

-52

-51

-50

-49

-48

-47

-46

-45

-44

-43

-42

-41

-40

-39

-38

-37

-36

-35

-34

-33

-32

-31

-30

-29

-28

-27

-26

-25

-24

-23

-22

-21

-20

-19

-18

-17

-16

-15

-14

-13

-12

-11

-10

-9

-8

-7

-6

-5

-4

-3

-2

-1

0

Selection/View

🔍

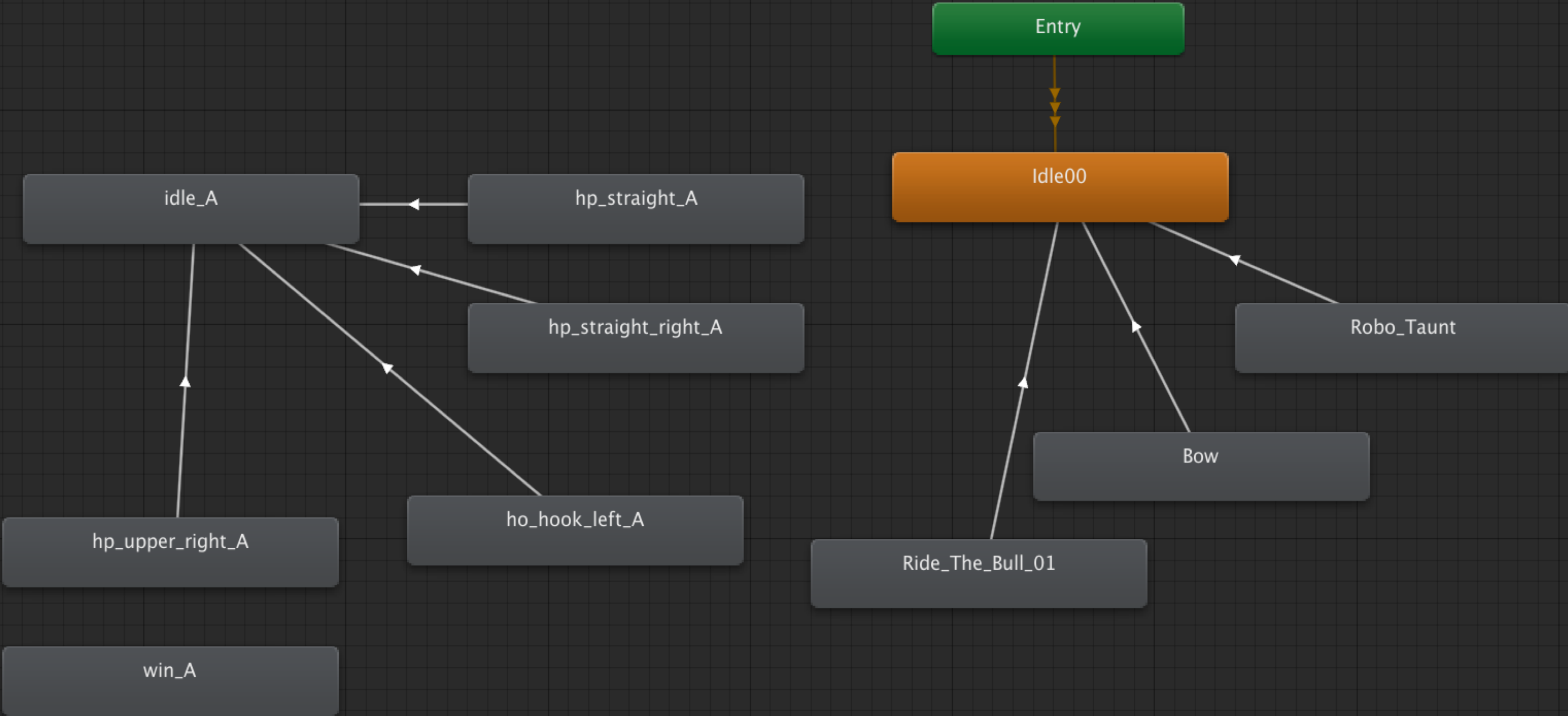
	Start	End	Duration
Selection	0:00.000	0:00.000	0:00.000
View	0:00.000	0:27.930	0:27.930

48000 Hz • 24-bit • Stereo

7.67 MB

0:27.930

488.27 GB free



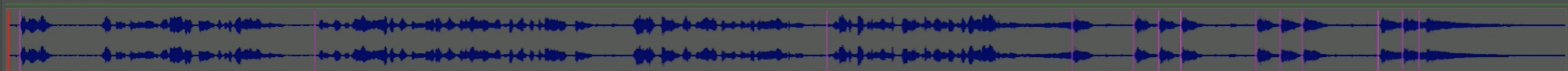
```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class MarkerNotifyAnim : MonoBehaviour
6  {
7      public Animator anim;
8
9      // Start is called before the first frame update
10     void Start()
11     {
12         Invoke("PlayAudio", 3f);
13     }
14
15     public void Notify(Fabric.EventNotificationType type, string eventName, object info, GameObject gameObject)
16     {
17         if (info != null)
18         {
19             if (type == Fabric.EventNotificationType.OnMarker)
20             {
21                 Fabric.MarkerNotificationData c = info as Fabric.MarkerNotificationData;
22                 if (c != null)
23                 {
24                     Debug.Log("Marker: " + c._label);
25                     anim.Play(c._label);
26                 }
27             }
28         }
29     }
30
31     public void PlayAudio()
32     {
33         Fabric.EventManager.Instance.PostEventNotify("animAudio", gameObject, Notify);
34     }
35 }
36

```



Animation Console Inspector



Subtitles

Waveform

Multitrack

Files

Favorites

📁

📄

🔍

🗑️

🔍

Name ↑	Status	Duration	Sample Rate	Channels	Bit Depth	Source Fo
> ++ GDC_RobotAnim_1.wav		0:27.930	48000 Hz	Stereo	24	Waveform
> ++ GDC_Rob_bttitles_1.wav		0:17.078	48000 Hz	Mono	24	Waveform

▶

📄

🔊

Media Browser

Effects Rack

Markers

Properties

🗑️

📄

🔍

🗑️

📄

🔍

🗑️

📄

🔍

Name	Start ↓	End	Duration	Type	Description
🔊 Thank you markers!	0:14.929		0:00.000	Cue	▼
🔊 Now you can understand...	0:12.788		0:00.000	Cue	▼
🔊 Like robot speak	0:11.443		0:00.000	Cue	▼
🔊 Or for made up languages	0:09.370		0:00.000	Cue	▼
🔊 The subtitles can be used ...	0:06.408		0:00.000	Cue	▼
🔊 Is the string for the subtitle	0:03.960		0:00.000	Cue	▼
🔊 The marker name	0:02.863		0:00.000	Cue	▼
🔊 Markers are commonly u...	0:00.034		0:00.000	Cue	▼

History

Video

📄

🔍

Open

0 Undo

🗑️

0.00.000

▶

⏸

⏮

⏪

⏩

⏭

🔴

📄

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

🔍

Levels

Selection/View

	Start	End	Duration
Selection	0:00.000	0:00.000	0:00.000
View	0:00.000	0:17.078	0:17.078

48000 Hz • 24-bit • Mono

2.35 MB

0:17.078

488.27 GB free

Editor: GDC_RobotSubtitles_1.wav

Mixer

Markers are commonly used for subtitles

The marker nar

Is the string for the subtitle

The subtitles can be used for localization

Or for made up languages

Like robot speak

Now you can understand me.

Thank you markers!

0:00.000

1.0

2.0

3.0

4.0

5.0

6.0

7.0

8.0

9.0

10.0

11.0

12.0

13.0

14.0

15.0

16.0

17.0

dB

-0.5

-1.0

-1.5

-2

-3

-4

-5

-6

-7

-9

-12

-15

-18

-24

-30

-30

-24

-18

-15

-12

-9

-7

-6

-5

-4

-3

-2

-1

0

0:00.000

0:00.000

0:17.078

0:17.078

Essential Sound

No Selection

Preset:


```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.UI;
5
6  public class MarkerNotifySubtitles : MonoBehaviour
7  {
8      public Text text;
9
10     // Start is called before the first frame update
11     void Start()
12     {
13         Invoke("PlayDialog", 3f);
14     }
15
16     public void Notify(Fabric.EventNotificationType type, string eventName, object info, GameObject gameObject)
17     {
18         if (info != null)
19         {
20             if (type == Fabric.EventNotificationType.OnMarker)
21             {
22                 Fabric.MarkerNotificationData c = info as Fabric.MarkerNotificationData;
23                 if (c != null)
24                 {
25                     Debug.Log("Marker: " + c._label);
26                     text.text = c._label;
27                 }
28             }
29         }
30     }
31
32     public void PlayDialog()
33     {
34         Fabric.EventManager.Instance.PostEventNotify("Subtitle", gameObject, Notify);
35     }
36 }
37

```



Animation Console Inspector

Regions (8)
|-- Markers are commonly used for subtitles
|-- The marker name
|-- Is the string for the subtitle
|-- The subtitles can be used for localization
|-- Or for made up languages
|-- The robot speak
Sync Regions With Tempo:
Audio Clip Tempo: 120

Num Of Loops: -1 [-1]
Infinite

Draw AudioClip Waveform:
Play AudioClip

Modulating SFX

Modulating SFX

Semi-tone by pitch value

-01 = 0.9438742	+01 = 1.059463
-02 = 0.8908986	+02 = 1.122462
-03 = 0.8408963	+03 = 1.189207
-04 = 0.7937004	+04 = 1.259921
-05 = 0.7491534	+05 = 1.33484
-06 = 0.7071066	+06 = 1.414214
-07 = 0.6674197	+07 = 1.498308
-08 = 0.6299603	+08 = 1.587402
-09 = 0.5946033	+09 = 1.681794
-10 = 0.5612308	+10 = 1.781798
-11 = 0.5297313	+11 = 1.88775
-12 = 0.4999997	+12 = 2.000001

Modulating SFX



Name ↑	Status	Duration	Sample Rate	Channels	Bit Depth	Source Format
> ++ MUS_Match3_1.wav		1:21.334	44100 Hz	Stereo	24	Waveform
> ++ MUS_Match3_2.wav		1:42.193	44100 Hz	Stereo	24	Waveform
> ++ MUS_Match3_3.wav		1:32.903	44100 Hz	Stereo	24	Waveform

Media Browser

Effects Rack

Markers

Properties

Name	Start ↓	End	Duration	Type	Description
End	1:21.334		0:00.000	Cue	
Bb mod	1:16.689		0:00.000	Cue	
D major modulation (+3 f...	1:05.076		0:00.000	Cue	
G modulation (-3)	0:51.141		0:00.000	Cue	
Start	0:00.044		0:00.000	Cue	
Tempo: 155.0	0:00.000		0:00.000	Cue	

History ≡ Video

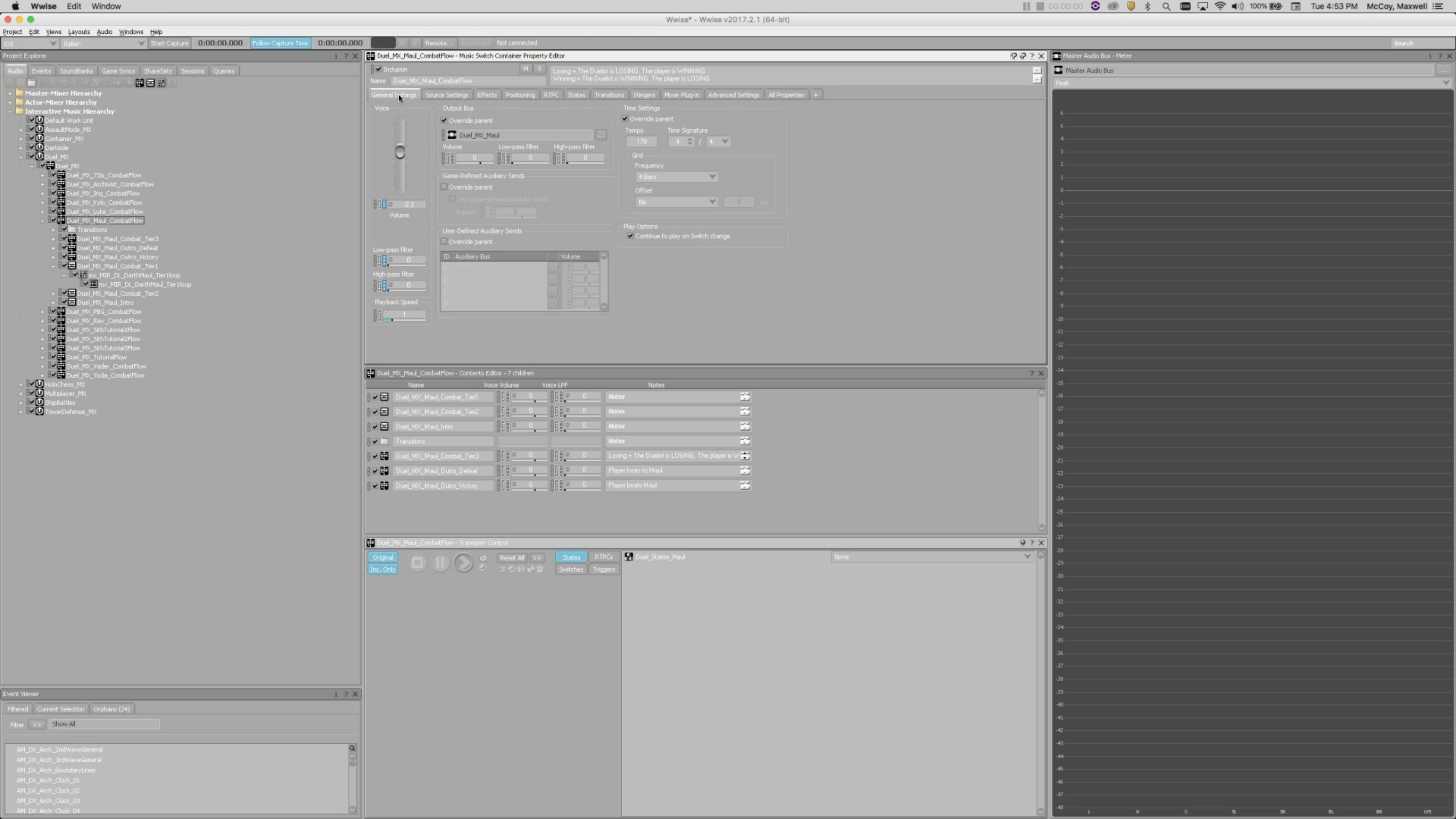
▶ Open

The screenshot displays a digital audio workstation (DAW) interface. At the top, the project name is 'MUS_Match3_1.wav' and the 'Mixer' tab is selected. The main workspace shows two stereo waveforms, one in red and one in blue, plotted on a grid. The time axis at the top ranges from 0:00.000 to 1:20.000. The bottom status bar shows the current time as 0:00.000 and various playback controls.

READY



Music Transitions



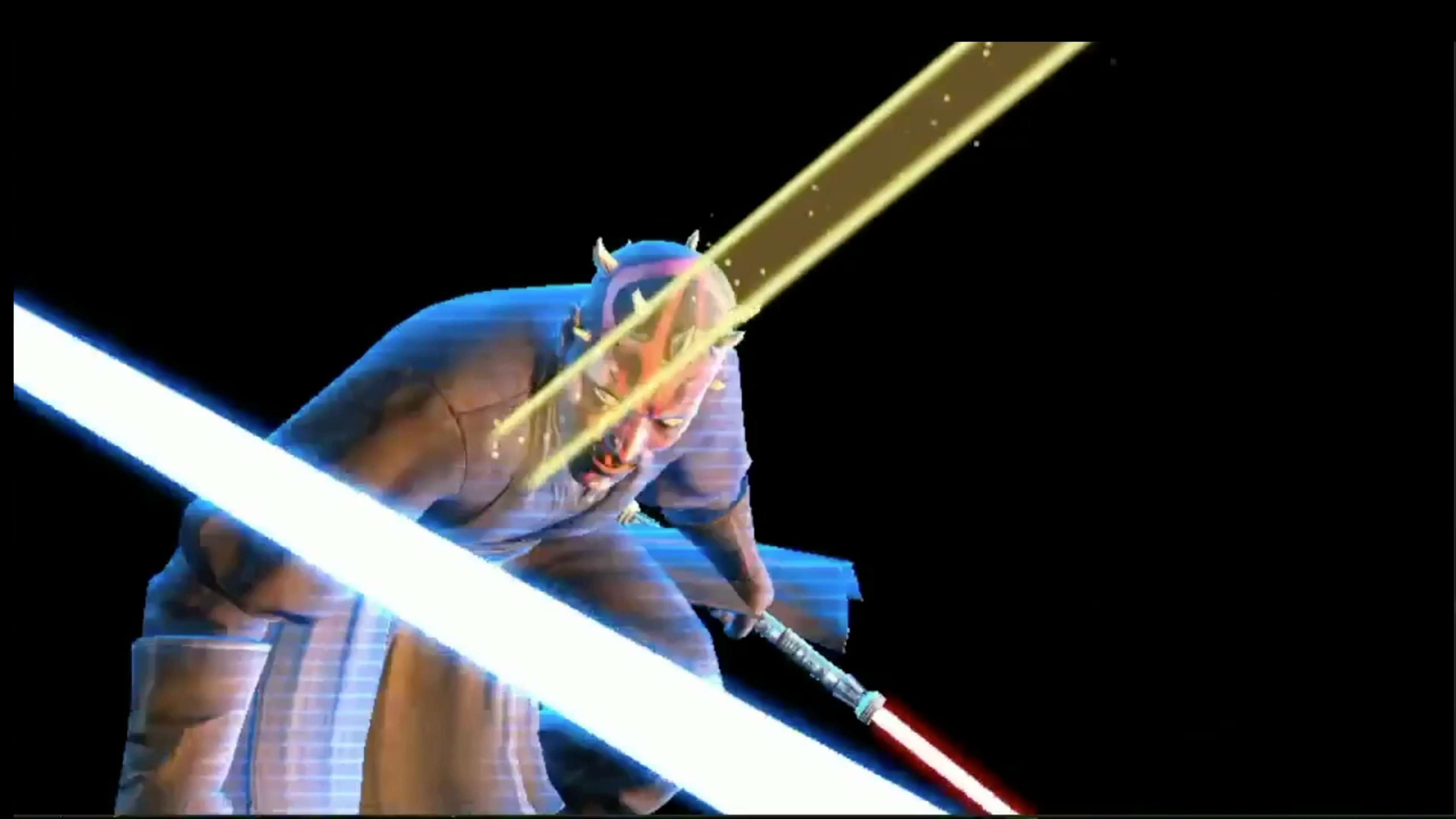
```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.Events;
5
6 public class WwiseMarkerNotify : MonoBehaviour {
7
8     public string mostRecentMarker;
9
10    AkCallbackManager.EventCallbackPackage wwiseEvent = null;
11
12
13    public void WwiseMusicEvent(string eventName)
14    {
15        AkSoundEngine.PostEvent(eventName, gameObject, (uint)AkCallbackType.AK_Marker, MarkerNotify, wwiseEvent);
16    }
17
18    public void MarkerNotify(object obj, AkCallbackType type, object info)
19    {
20        AkMarkerCallbackInfo MarkerCallbackInfo = (AkMarkerCallbackInfo)info;
21        mostRecentMarker = MarkerCallbackInfo.strLabel;
22        Debug.Log("Most Recent Marker " + mostRecentMarker);
23
24        if(mostRecentMarker == "MarkerForAction")
25        {
26            //Do Action
27        }
28    }
29 }
30

```


To Begin





Markers can be used for
anything that you can
dream up.

The GDC logo is centered at the top of the slide. It consists of the letters "GDC" in a bold, white, sans-serif font. The letters are positioned over a large, solid red triangle that points downwards. The background of the entire slide is a dark blue. There are several decorative elements: a small red square with three white squares to its right in the top-left corner; a small red square with three white squares to its right in the top-right corner; a small red square with three white squares to its right in the bottom-left corner; and a small red square with three white squares to its right in the bottom-right corner. Additionally, there are thin red lines forming large triangles that point towards the center of the slide, and a few small red diamonds scattered around.

Thank you!

Questions?

GAME DEVELOPERS CONFERENCE

MARCH 18–22, 2019 | #GDC19