



Unite Seoul 2019



Blending Gameplay and Storytelling in Timeline: Improvements in 2019

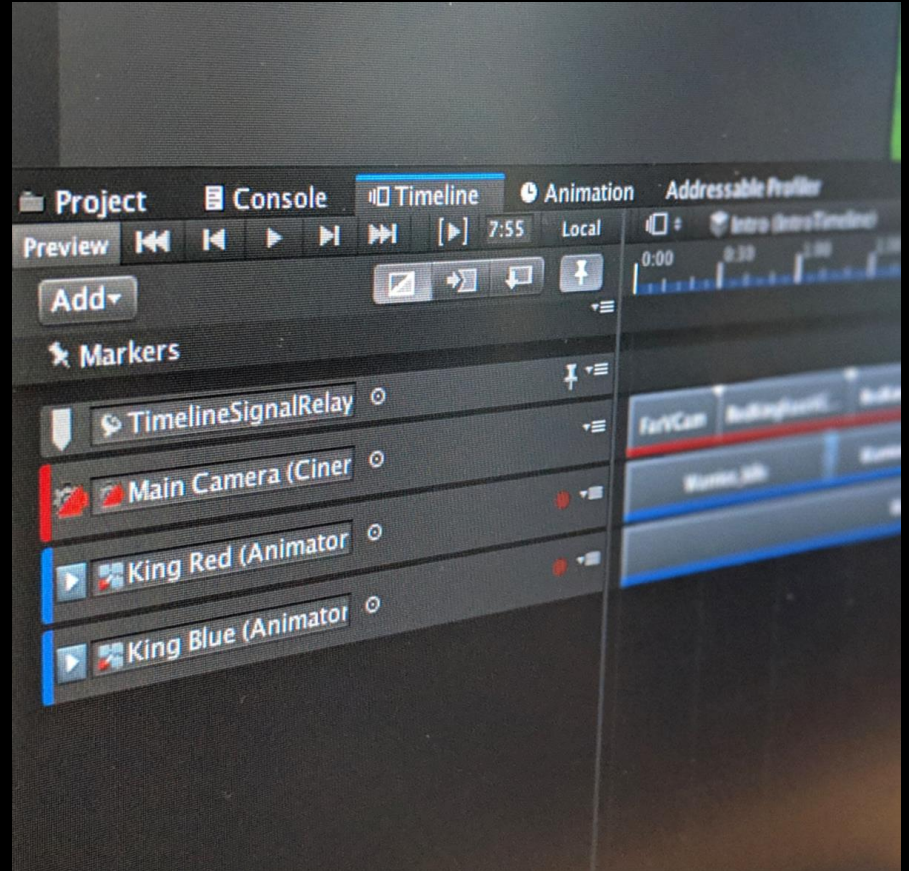
Ciro Continisio - Technical Evangelist

← unity

Unite
Seoul
2019

Timeline improvements 18.3 & 19.1

- Markers and Signals
 - Custom markers
- Animator track offset (improved)
- Audio track controls (improved)
- Control track nesting



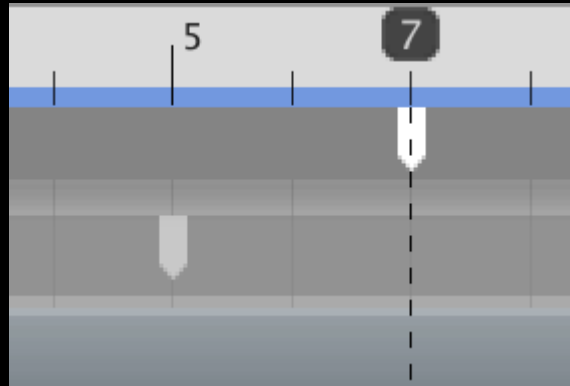
Markers and Signals



Markers

Markers give the ability to mark an **instant** in time on a Timeline.

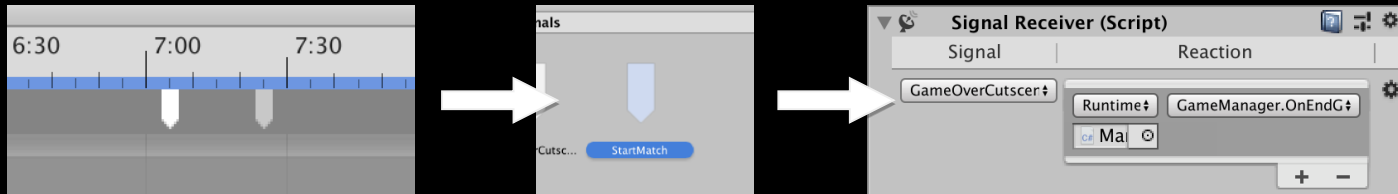
They are **opposed to clips**, which represent a continuous segment of time (and have a start, a duration and an end).



Signals

Signals are a specific type of marker, and require 3 parts to work:

- **Signal Emitter** on the Timeline, which emits the notification
- **Signal (Asset)** which is the channel through which the notification is broadcasted
- **Signal Receiver (Component)**



The background is an abstract, fluid composition of colors. It features deep, dark blue and teal tones that swirl and flow together, creating a sense of movement and depth. Interspersed within these darker areas are bright, glowing orange and yellow elements, which appear as if they are catching light or reflecting off a surface. The overall effect is ethereal and dynamic, resembling a microscopic view of a liquid or a digital simulation of fluid dynamics.

Let's see it in the editor

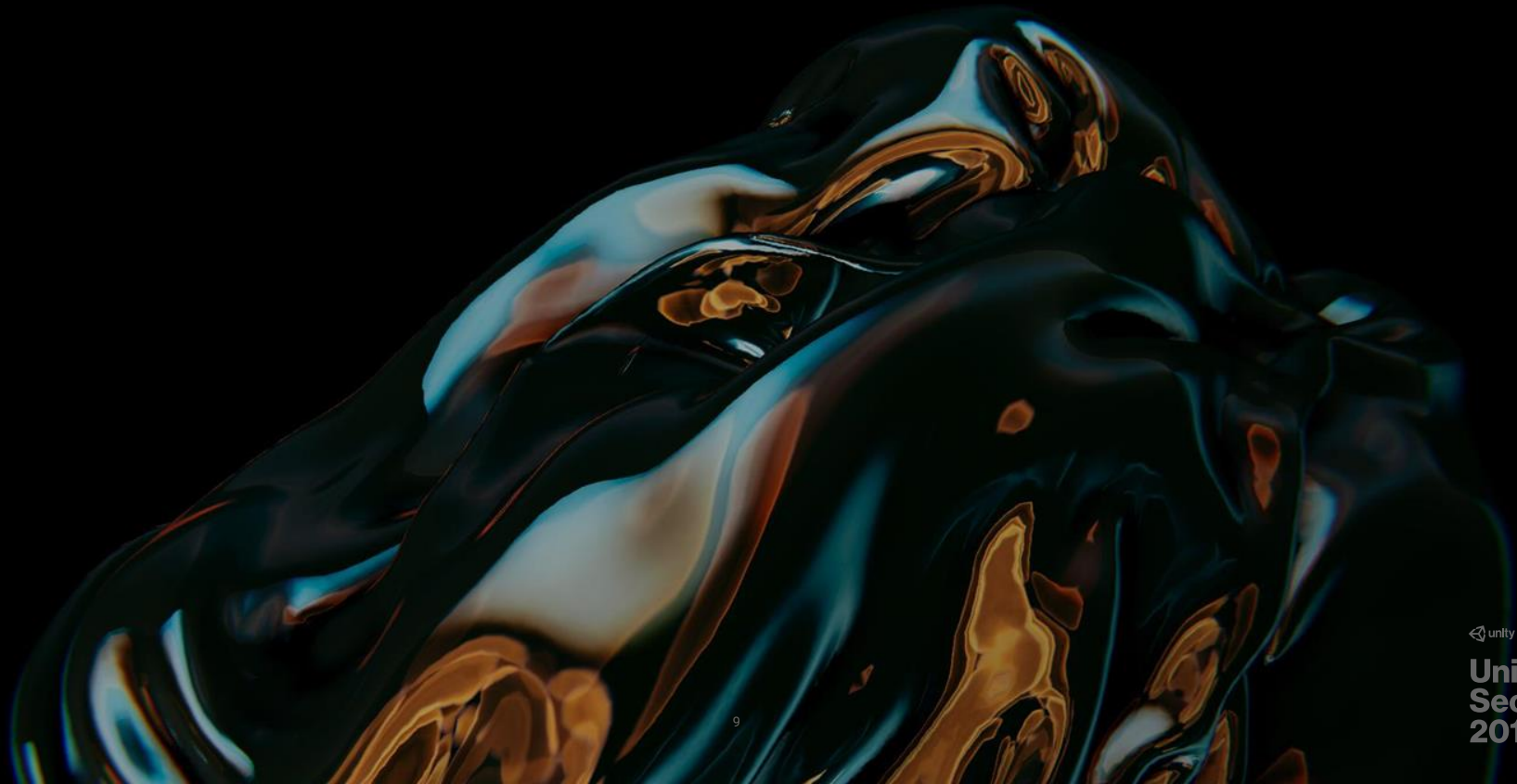
Signals - Different ways to use them

You can put markers and Signals in three different places:

1. On the Timeline's own Markers track
2. On a dedicated Signals track
3. On any track that supports a binding



Custom markers



Creating your custom marker

To create a simple marker, you need 1 script:

- One which defines the marker, inheriting from Marker

To make the marker trigger code, you need 2:

- **Marker**: Inherits from Marker, implements INotification
- A **MonoBehaviour**: implements INotificationReceiver

The background is an abstract, fluid composition of colors. It features deep, dark blue and teal tones that swirl and flow, creating a sense of movement. Interspersed within these darker areas are bright, glowing orange and yellow elements, which appear as if they are catching light or reflecting off a surface. The overall effect is ethereal and dynamic.

Let's see it in the editor

CardMarker: the marker on the Timeline

```
1  [Serializable, DisplayName("Card Marker")]
2  public class CardMarker : Marker, INotification
3  {
4      public CardData card;
5      public Vector3 position;
6      public Placeable.Faction faction;
7
8      //required by INotification but we're not actually using it
9      public PropertyName id { get { return new PropertyName(); } }
10 }
11
12
13
14
15
```

CardPlayerBridge: receives the notification

```
1  public class CardPlayerBridge : MonoBehaviour, INotificationReceiver
2  {
3      public GameManager gameManager; //public reference
4
5      //will ask the manager to play a Card
6      public void OnNotify(Playable origin, INotification notification, object context)
7      {
8          CardMarker cm = notification as CardMarker;
9
10         if(cm != null)
11             gameManager.UseCard(cm.card, cm.position, cm.faction);
12     }
13 }
14
15
```

Other improvements

Extra resources

Unity Blog | Ideas on creative uses for Timeline with custom clips

bit.ly/CreativeTimelineBlogPost

YouTube | Same as above but in video format


bit.ly/CreativeTimelineYoutube

Unity Blog | A guide to get started on creating custom tracks/clips

bit.ly/GetStartedScriptingTimeline

Thank you! Questions?

Ciro Continisio - Technical Evangelist

 @CiroContns

 unity

Unite
Seoul
2019