



# Unityにおける アセットの非同期読み込みについて

発表日：2019年 10月 10日

対象：後世の新卒向け

発表者：鈴木 健太

# 自己紹介

- ▶ 名前：鈴木 健太
- ▶ 部署：GS部→AG部
- ▶ 入社：2019/04/01(新卒)
- ▶ 業種：エンジニア

よろしく  
お願いいたします。



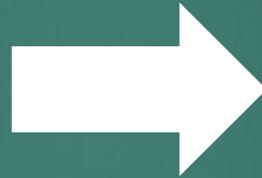
- ▶ テーマ：Unityにおけるアセットの非同期読み込みについて
- ▶ 対象：後世の新卒向け
- ▶ 設定理由
  - 学生では触れる機会が少ないが製品としての制作では必須
  - 後世の新卒が同じ箇所で躓きにくくなるように
- ▶ アセット = ゲーム内で使用する画像などのデータ

# 何に躓いたのか

▶ アセットバンドル

▶ コルーチン

▶ コールバック



なにそれ、  
おいしいの？



## なぜ躓いたのか

- ▶ 基本中の基本がわかっていること前提の記事が多い
- ▶ 概要を知りたいのにより深いところまで解説されている
  - 欲しい情報以外の情報がたくさんありすぎてパンクする

**時間がもったいない！**

アセットバンドル

コルーチン

コールバック

アセットバンドル ←

コルーチン

コールバック

- ▶ アセットの格納・読み込みのシステム
  - 複数のアセットを1つの塊(アセットバンドル)に圧縮できる
  - アセットバンドルを読み込むとアセットが使用できる
  - アセットの読み込みは別途必要
  
- ▶ アセット圧縮(アセットバンドル化)の基準は様々
  - 例1) 1つのアセットバンドル = 1つのアセット
  - 例2) キャラクターごと(画像、ボイス、アニメーションなど)

## アセットバンドルを例えると…

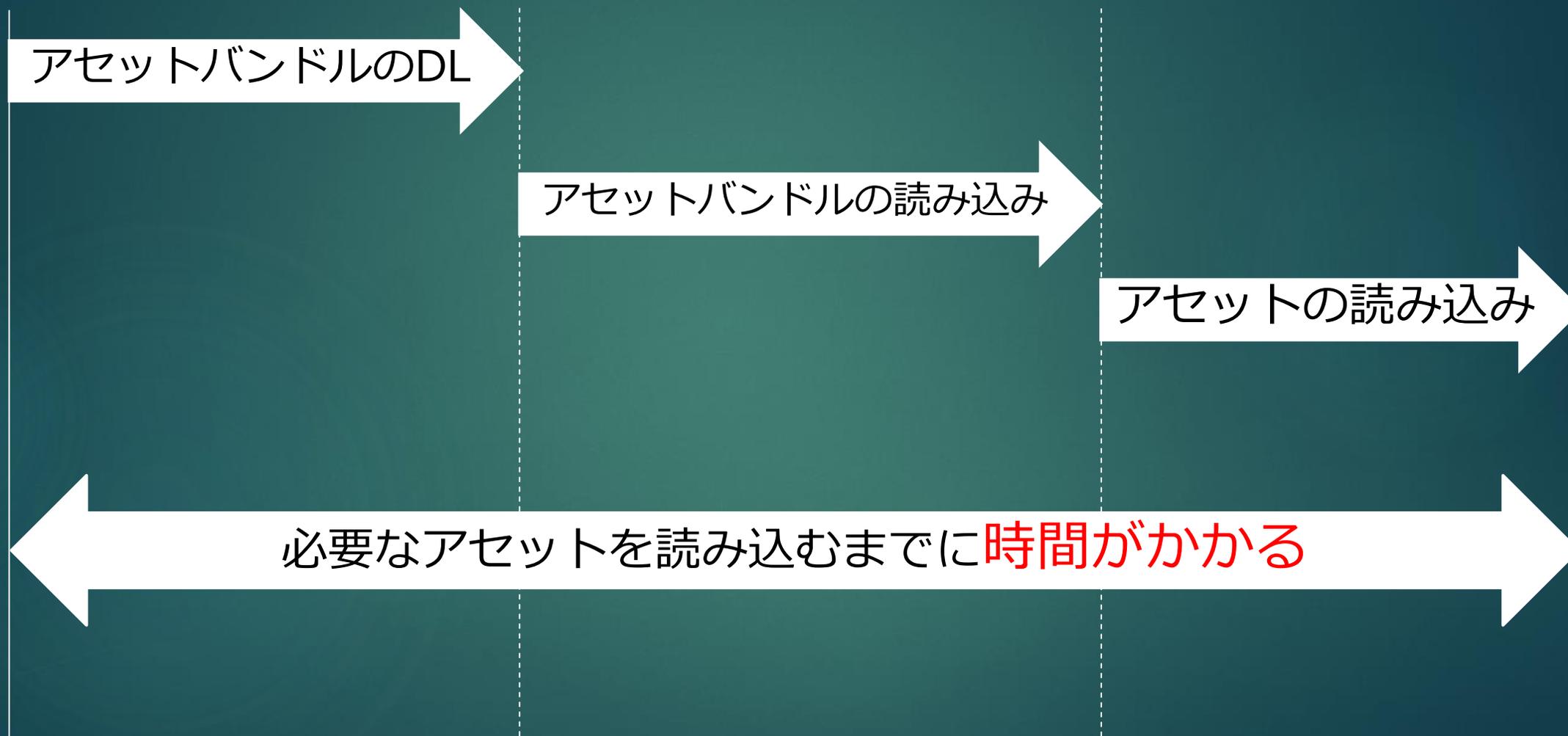
アセットバンドル	zipフォルダ
アセットの圧縮	フォルダ圧縮
アセットバンドルの読み込み	フォルダ解凍
アセットの読み込み	

## アセットバンドルはどこに置く？

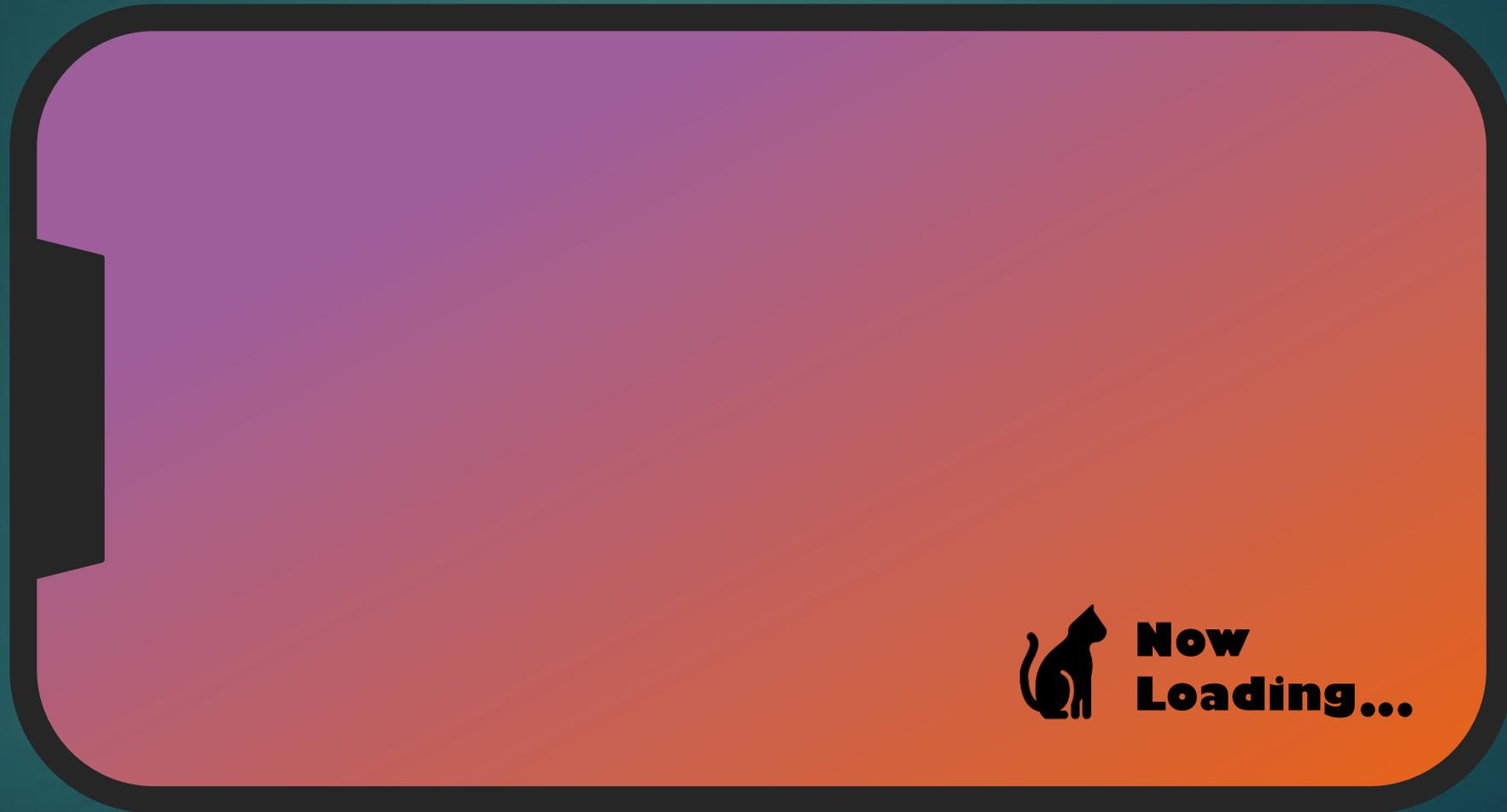
- ▶ アセットバンドルはアプリ本体 or サーバーに置く
- ▶ サーバーに置いた場合
  - デメリット：使うときにダウンロードする必要がある
  - メリット：アプリ本体の容量が小さく済む
- ▶ アプリ本体に置いた場合
  - サーバーに置いた場合と逆



# アセットが使用可能になるまで



# ロード画面のイメージ



- ▶ メイン処理はロード画面を表示
- ▶ メイン処理の裏で時間のかかる処理を行う



非同期処理

アセットバンドル  
コルーチン }  
コールバック }

アセットバンドル

コルーチン ←

コールバック

- ▶ メイン処理から独立して処理が実行される
  - つまり非同期処理
- ▶ レジューム機能を持つ
  - 一連の処理の中で一旦停止し、その箇所から再開できる
  - 一連の処理をフレームを跨いで実行できる

※一旦停止 → 「処理を進めず何もしない」という処理をする

### ▶ 時間経過を用いるイベント

- 定期的に処理を実行

例) フェード処理 → 1フレームごとに $\alpha$ 値を更新

### ▶ 重い処理を実行

- 複数フレームに処理を分割して処理落ちを防ぐ

例) アセット読み込み → アセットバンドルのダウンロード、  
アセットバンドルの読み込み、  
アセットの読み込みを別フレームで処理

アセットバンドル

コルーチン

コールバック ←

- ▶ 別のメソッドに呼んでもらうためのメソッド(処理の塊)
  - 処理の塊を引数として渡すことが多い
- ▶ 呼び出し側で好きなタイミングで処理の塊を呼べる
  - 処理Aと処理Bの間で呼べるようにする
  - 非同期処理後に実行したい処理を呼ぶ

# アセットの非同期読み込みの流れ

1. ロード画面の表示開始

2. アセットバンドルのDL

3. アセットバンドルの読み込み

4. アセットの読み込み

5. ロード画面の表示終了

非同期処理

コルーチン

コールバック

## ▶ アセットバンドル

- アセットの格納と読み込みのシステム ≡ zipフォルダ
- アプリの容量を小さくすることも可能

## ▶ コルーチン

- 非同期処理ができるメソッド + レジューム機能

## ▶ コールバック

- 処理の塊を好きな時に別のメソッドで呼ぶ手法

非同期読み込みを使いこなして  
快適なゲーム制作を！

- ▶ 今更誰も教えてくれない、Unityにおけるアセット読み込みについての基礎知識 (2019年9月2日 最終確認)  
<https://qiita.com/k7a/items/df6dd8ea66cbc5a1e21d>
- ▶ AssetBundleを完全に理解する (2019年9月2日 最終確認)  
<https://qiita.com/k7a/items/d27640ac0276214fc850>
- ▶ アセットバンドルの基礎 (2019年9月2日 最終確認)  
<https://unity3d.com/jp/learn/tutorials/topics/best-practices/assetbundle-fundamentals>
- ▶ アセットバンドルに複数のアセットを格納して取り出す (2019年9月2日 最終確認)  
[https://www.shibuya24.info/entry/multi\\_ab](https://www.shibuya24.info/entry/multi_ab)

- ▶ コルーチン - Unity マニュアル (2019年9月2日 最終確認)  
<https://docs.unity3d.com/ja/current/Manual/Coroutines.html>
- ▶ 【Unity】コルーチンってなんなのなの？って時に読む記事【解説】 (2019年9月2日 最終確認)  
<https://ekulabo.com/whats-coroutine>
- ▶ Unityのコルーチンを分かりやすく説明してみる。  
(2019年9月2日 最終確認)  
<https://qiita.com/ritsuka/items/0263ece6594b47455309>

## 参考文献(コールバック)

- ▶ 「分かりそう」で「分からない」でも「分かった」気になれる  
IT用語辞典, コールバック関数(callback function)  
(2019年9月2日 最終確認)  
<https://wa3.i-3-i.info/word12295.html>
- ▶ 【Unity】 ActionとEventを勉強した手順  
(2019年9月2日 最終確認)  
<https://hk-ryukyu.club/hideto/archives/49#toc4>
- ▶ C# 備忘録 コールバックについて  
(2019年9月2日 最終確認)  
<https://qiita.com/r-ngtm/items/6e5e5fbe6a58df3b1bc8>