

UnrealEngineでアクタの単体テストをやってみた

@mechamogera

自己紹介

- mechamogera
- 主にQiitaで活動 : <https://qiita.com/mechamogera>
- 包丁で指を切って4針縫った
- 謎のあせも発症
- 昨日花粉症発症

UnrealEngineとアクタ

- UnrealEngine

- EpicGamesが提供する3Dプログラミングのプラットフォーム

- アクタ

- マップに配置するオブジェクトの基本単位、ベースクラス

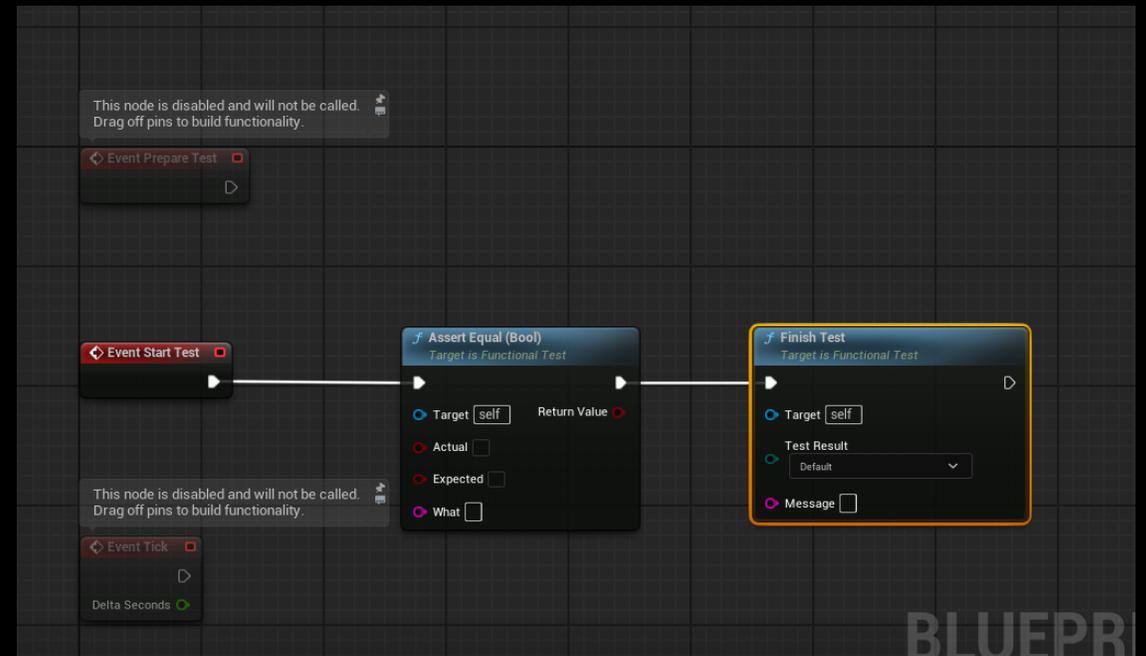
- キャラクターやライト、カメラなどの様々なアクタが存在

自動テストの背景

- テスト対象：VRマルチプレイのコミュニケーションアプリ
 - VR：グラフィカルな要素多い
 - 入力：コントローラーと手の動き
 - マルチプレイ：複数接続が必要
- UnrealEngine
 - テストフレームワーク：マルチプレイ/モック/スタブ非サポートでテストの自動化が困難
- => ただ、ロジック部分は自動テストしたい
感覚的に一番バグるのがネットワークロジック回り
変数の設定で限定的にマルチプレイも確認可能

UnrealEngineのFunctional Test

- UnrealEngineのテストフレームワーク
 - [Unreal Engine の機能テスト | Unreal Engine 5.5 ドキュメンテーション | Epic Developer Community](#)
- FunctionalTestクラスまたはその子クラス(C++ or Blueprint)をLevelに配置することでテストが可能

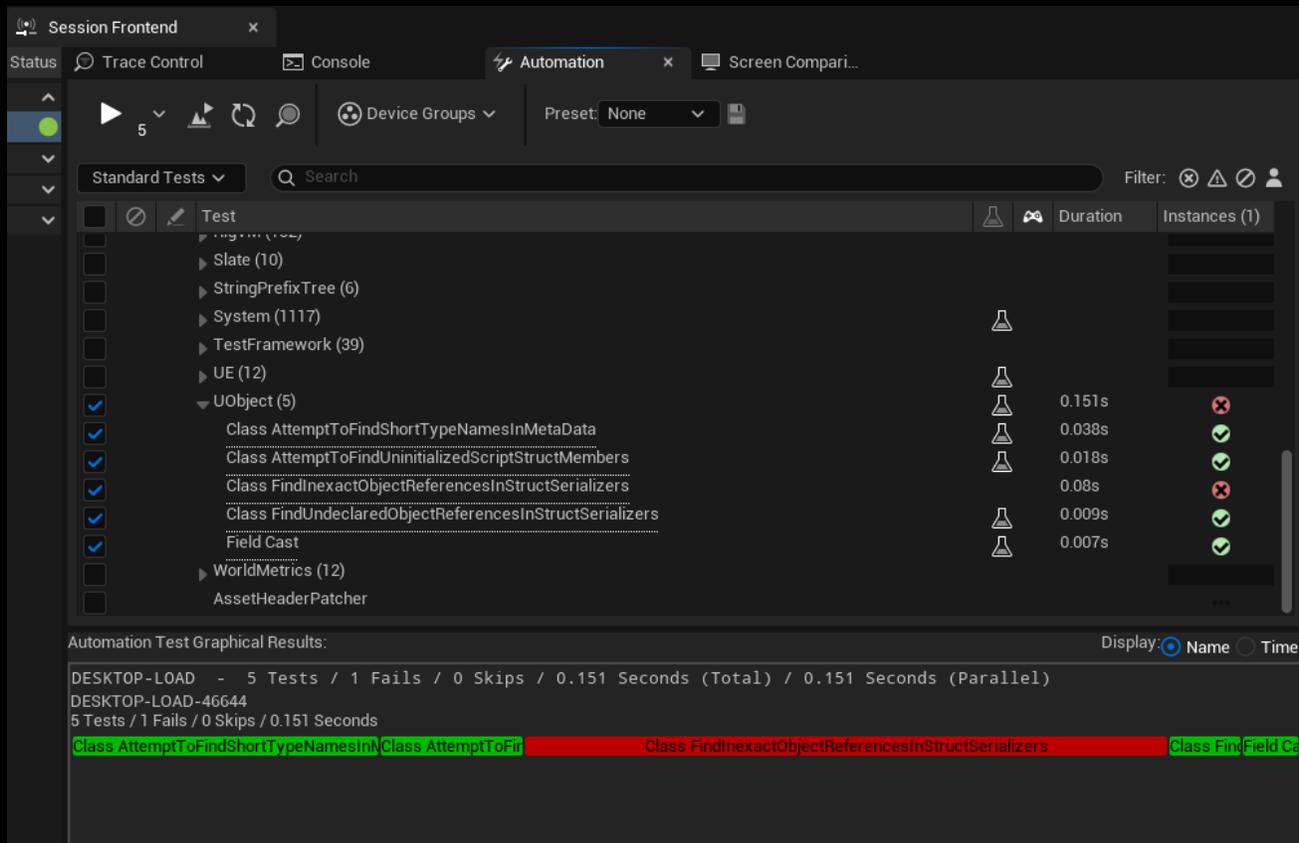


アクタの単体テストの実装フロー

- 対象のアクタクラスを継承してテスト用の子クラスを作成
- メソッド、カスタムイベントをオーバーライドしてコストの重い処理を無効化
 - サーバーにファイルをアップロードする
 - ファイルを読み込む
- 依存するアクタを継承するかインターフェースを実装してインスタンスを置き換える口を用意して置き換え
- **FunctionalTest**を継承してテスト用の子クラスのテストを実装
- 作成したクラスのインスタンスをLevelに配置

テストの実行方法

- UnrealEngine Editorから実行可能



CICDでのテストの実行

- コマンドラインからテスト実行可能
UE4Editor.exe -NullRHI -nosplash [Project].uproject -
ExecCmds="Automation RunTests [テスト];Quit"
 - -NullRHI：ウィンドウを表示しない
 - -nosplash：起動時のスプラッシュ画面を表示しない
 - -ExecCmds：起動時にコンソールコマンドを実行
 - 指定したテストを実行して終了
- 本当はテスト→パッケージビルドにしたいがgit cleanしているのでクリアされてしまうのでパッケージビルド→テストで実施
ビルド→テスト→パッケージビルドだと遅い？

パッケージにテストアセットを含めないようにする

- テスト用のアセットを含めるとパッケージのサイズが大きくなる
- パッケージングに含めない2つの方法
 - プラグイン形式：テスト関連コンテンツをプラグイン形式にしてパッケージングの際はプラグインを外す
 - Cookしない：Packagingの設定でDirectories to never cookでテストのルートフォルダを指定する

テストをマップに追加するスクリプト

- テストを用意するたびにマップに配置する必要あり
- 手動が面倒なのでメニューから一発でできるようにしてみた
- UnrealEngineで選択したアセットを右クリックメニュー1発でマップに配置する #UnrealEngine5 - Qiita
 - テスト用のマップを開いて選択したアセットのインスタンスがなければ作成して保存

テストをやってみて

- テスト実装の手間はあるがエラーケースのテストがしやすい
- エラーログは出ているがシレっと通っているケースが分かる
- 単体テストでは限界がある
- RPCの実行確認などはできないのでネットワークのテストほしい
- 手元環境で成功するがCICD環境で失敗するケースがあった