

Unity MIDI Plugin — ドキュメント目次

このドキュメントでは、[Assets/MIDI](#) の内容、ランタイム API の構造、および Unity で MIDI 1.0、MIDI 2.0 (UMP)、MPE、および関連するトランスポートを使用する方法について説明します。

言語

- [English](#)
- [中文\(简体\)](#)

目次

はじめに

- [スタートガイド \(インストール、初期化、送受信\)](#)
- [ビルド後の処理 \(PostProcessing\) とスクリプト定義シンボル](#)
- [プラットフォームと制限事項](#)

コア API

- [MIDI 1.0 \(MidiManager\)](#)
- [MIDI 2.0 / UMP \(Midi2Manager\)](#)
- [MPE \(MIDI Polyphonic Expression\)](#)
- [SMF / シーケンシング \(jp.kshoji.midisystem\)](#)

トランスポートと統合

- [トランスポートとプラットフォーム](#)
- [アプリ間 MIDI 一クロスプラットフォームの注意点 \(Android, iOS/macOS, Linux\)](#)
- [Maestro / MPTK 統合 \(仮想デバイス + アダプター\)](#)

プロジェクト構成の注意点

- [仮想デバイスとイベント注入](#)
- [MIDI-CI / 機能ネゴシエーション](#)
- [エディタとライフサイクルに関する注意点](#)
- [組み込み済みサードパーティモジュール](#)

サンプルとリファレンス

- [サンプル](#)
- [動作確認済みデバイス](#)
- [連絡先 / サポート](#)
- [更新履歴](#)

ディレクトリ構造 (Assets/MIDI)

- **Plugins/**
プラットフォームごとのネイティブ（および WebGL JS）プラグイン
(Android/iOS/macOS/Linux/WSA/WebGL)。
- **Scripts/**
メインの C# ランタイム:
 - `MidiManager.cs` (MIDI 1.0)
 - `Midi2Manager.cs` (MIDI 2.0 / UMP)
 - プラットフォームプラグイン: `MidiPlugin.*.cs`, `Midi2Plugin.*.cs`
 - イベントハンドラインターフェース: `IMidi*EventHandler`, `IMidi2*EventHandler`
 - MPE: `MpeManager.cs`, `IMpeEventHandler.cs`
 - MIDI-Cl: `MidiCapabilityNegotiator.cs`
 - 仮想デバイス: `MidiManager.VirtualDevices.cs`
- **Scripts/midisystem/**
標準 MIDI ファイル (SMF) リーダー/ライター + シーケンシングモデル (`Sequence`, `Track`, メッセージ)。
- **Scripts/UmpSequencer/**
UMP シーケンスユーティリティ (クリップ/コンテナの読み書き、シーケンサー、SMF↔UMP コンバータ)。
- **Scripts/RTP-MIDI-for-.NET/**
RTP-MIDI 実装 (組み込みモジュールとそのドキュメント)。
- **Scripts/UdpMidi2Discovery/**
UDP MIDI 2.0 ワークフロー用のディスカバリ依存関係。
- **Samples/**
サンプルシーンとスクリプト。

概念と用語

- **DeviceId:** MIDI エンドポイントを参照するために API 全体で使用される文字列識別子。
- **Group:** MIDI 2.0 グループインデックス (0–15)。一貫性のために MIDI 1.0 API でも使用されます。
- **Channel:** MIDI チャンネル (0–15)。一部の API は 0 オリジンのチャンネル番号を使用することに注意してください。
- **UMP (Universal MIDI Packet):** このプロジェクトでは `uint[]` ワードとして表現される MIDI 2.0 パケット形式。

クイックスタート・チェックリスト

1. 必要な機能セットを決定します:
 - MIDI 1.0 イベントと送信: `MidiManager`
 - MIDI 2.0 / UMP パースと送信: `Midi2Manager`
 - Android アプリ間 MIDI: [アプリ間 MIDI](#) を参照
 - MPE 管理: `MpeManager` (MidiManager 上で動作)
2. 1つ以上のイベントハンドラインターフェースを実装します。
3. ハンドラオブジェクトをマネージャーに登録します。
4. マネージャーを初期化します（またはシーン内に存在することを確認します）。
5. `Assets/MIDI/Samples/Scenes` にあるサンプルシーンを使用してテストします。