Unity MIDI Plugin — Documentation Index

This documentation describes the contents of `Assets/MIDI`, how the runtime API is structured, and how to use MIDI 1.0, MIDI 2.0 (UMP), MPE, and related transports in Unity.

Languages

- 日本語
- 中文(简体)

Contents

Directory structure (Assets/MIDI)

- `Plugins/`
  Native (and WebGL JS) plugins per platform (Android/iOS/macOS/Linux/WSA/WebGL).
- `Scripts/`
  Main C# runtime:
  - `MidiManager.cs` (MIDI 1.0)
  - `Midi2Manager.cs` (MIDI 2.0 / UMP)
  - platform plugins: `MidiPlugin.*.cs`, `Midi2Plugin.*.cs`
  - event handler interfaces: `IMidi*EventHandler`, `IMidi2*EventHandler`
  - MPE: `MpeManager.cs`, `IMpeEventHandler.cs`
  - MIDI-CI: `MidiCapabilityNegotiator.cs`
  - virtual devices: `MidiManager.VirtualDevices.cs`
- `Scripts/midisystem/`
  Standard MIDI File (SMF) reader/writer + sequencing model (`Sequence`, `Track`, messages).
- `Scripts/UmpSequencer/`
  UMP sequence utilities (clip/container read/write, sequencer, SMF↔UMP converter).
- `Scripts/RTP-MIDI-for-.NET/`
  RTP-MIDI implementation (embedded module + its own docs).
- `Scripts/UdpMidi2Discovery/`
  Discovery dependencies for UDP MIDI 2.0 workflows.
- `Samples/`
  Example scenes and scripts.

Concepts & terminology

- **DeviceId**: A string identifier used throughout the API to refer to a MIDI endpoint.
- **Group**: MIDI 2.0 group index (0–15). Many MIDI 1.0 APIs accept `group` for consistency.
- **Channel**: MIDI channel (0–15). Note that some APIs use 0-based channel numbers.
- **UMP (Universal MIDI Packet)**: MIDI 2.0 packet format represented as `uint[]` words in this project.

Quick start checklist

1. Decide which feature set you need:
   - MIDI 1.0 events and sending: `MidiManager`
   - MIDI 2.0 / UMP parsing and sending: `Midi2Manager`
   - Android Inter-App MIDI: see Inter-App MIDI
   - MPE management: `MpeManager` on top of `MidiManager`
2. Implement one or more event handler interfaces.
3. Register handler objects with the manager.
4. Initialize the manager (or ensure it’s present in the scene).
5. Test using the sample scenes in `Assets/MIDI/Samples/Scenes`.