

# 快捷键操作

Ctrl+L 打开或关闭日志窗口

Ctrl+D 日志级别在 Debug 和 Info 之间切换, 调试使用, 并不保存日志级别参数

Ctrl+R 在不关闭应用程序情况下, 重新加载 MeidaContents.Config 配置文件

Ctrl+F 窗体状态, 在最大化和还原的状态之间切换

Ctrl+S 窗体样式, 有在边框和无边框之间切换

Ctrl+T 窗体置顶, 在置顶和非置顶之间切换

数字键 0~9, 对应调用 LoadItem(int id) 指定节点 ID

方向键 Down/Right, LoadItem(int id) 下一个节点 Item

方向键 UP/Left LoadItem(int id) 上一节点 Item

空格/回车, 播放/暂停之间切换

Esc 键, 退出程序

## 配置说明

注意: 可以删除不使用的或不必要的配置项或属性, 不影响程序的运行, 但会影响其运行功能

【MediaPalyerPro.exe.Config】为应用程序运行时初使需要的配置

【MediaContents.Config】为媒体内容配置

【节点 Item】: 可理解为项或页(Page), 或组合显示对象, Item 的一级子节点(Window, BackgroundGroup, BackgroundPlayer, BackgroundButtons, MiddleGroup, MiddlePlayer, MiddleButtons, ForegroundGroup, ForegroundPlayer, ForegroundButtons)为显示对象, 可配置其相应的属性值

**Window:** 主窗体, 主要应用于在不同的视频场景, 或不同的内容页面切换不同的窗体状态, 如果在整体应用过程中不需要发生变化, 可在 "MediaPlayerPro.exe.Config" 只配置一次

**Group:** 显示对象容器或组件, 其中包含了 **Player**、**Buttons** 一组显示对象; Middle、Foreground 默认为隐藏状态

**Player:** 可以配置视频、图片和音频文件, 默认为居中位置; 如果要自定义位置, 可设置为左顶上角位置, 在设置 Margin 属性

**Buttons:** 按钮的集合, 每次 LoadItem(int id) 都会先清空之前的 Buttons 集合

注意: 1.Background、Middle、Foreground 的概念, 不应理解为字面意思, 它只是三组动态显示对象容器组件, 每组都可以动态配置媒体文件、按钮、交互动作(方法)等等, 它们的配置参数都相同; 如果三组都配置使用, 将会导致配置复杂化, 而且要考虑硬件的配置, 是否支持三组视频播放器内核的渲染压力

2.串口对象的 Key 只访问前缀

3.配置是按节点、行顺序执行

### 【属性】

**显示对象常用基本属性:** Width="Number", Height="Number";

IsEnabled="False/True", Visibility="Visible/Hidden", Opacity="0.0~1.0";

HorizontalAlignment="Left/Center/Right/Stretch", VerticalAlignment="Top/Center/Bottom/Stretch";

Margin="Left,Top", Stretch="None/Fill/Uniform/UniformToFill";

**Player** 特殊属性(Player 也是显示对象): AutoOpen="True/False" 设置 Url 属性后自动打开调用 Open(Link,null) 媒体文件, 只是打开媒体文件并不会播放, 默认为 True;

OpenAndPlay="True/False", 调用 Open(Link,null) 后是否自动播放, 如果为 False 则需要手动再次调用 Play() 才播放, 默认为 True;

Loop="True/False", 视频循环播放, 可实现头尾帧无缝连接循环播放, 默认为 True;

Volume="0.0~1.0", 音量属性, 默认为 0.8;

支持带 Alpha 通道的视频文件, 图片文件, 音频文件;

以及其它继承属性;

**Button** 为可视对象, 其属性 Name 在当前项 (Item) 中, 不能同名; 按钮有 Background(鼠标点击后显示) 和 Foreground(默认为显示, 鼠标放上去后透明 0.8), 可设置为图片、颜色或隐藏效果等, 按原 XAML 布局编码解析灵活性及大;

**Action** 动作(方法)节点属性: TargetObj/TargetKey, Method, Params(多个参数以 ',' 分隔), 一个 Action 节点中 TargetObj 与 TargetKey 只可存在一个, TargetKey 为通信连接对象, 参考 MediaPlayerPro.exe.Config, TargetObj 为显示对象; 支持属性设置: TargetObj/TargetKey, Property, Value;

## 【方法 Action】

**Window:** 可访问的方法, 除 API 中自带的方法外, 还包括 LoadItem(int id), LoadNextItem(int timer, int id), CallButtonEvent(int id, string layerButtonsName, string buttonName), Sleep(int ms), PlayPause(), NextItem(), PrevItem(), NextNode(), PrevNode(), SetVolume(float volume), VolumeUp(), VolumeDown() 等; 窗体的 SetVolume(float), VolumeUp(), VolumeDown() 是全局, 针对所有 Player; 注意: NextItem(), PrevItem() 是播放临近的 ID 项, 对 ID 值递增或递减, NextNode(), PrevNode() 是播放临近的 XML 节点, 上一个节点或下一个节点。

**Player:** 除 API 自带方法外, 还包括 Open(Link,null), Play(), Pause(), SetLoop(bool loop), SetVolume(float volume), VolumeUp(), VolumeDown(), SeekFastMilliSecond(int ms), Close() 等

**TCP/UDP/SerialPort 连接对象**(TargetKey, 参考 MediaPlayerPro.exe.Config), 除 API 自带的方法外, 还包括 SendBytes(byte[] bytes), SendMessage(string message) 等

**Modbus 连接对象**, TurnSingleCoilAsync(byte address, ushort startAddress), TurnMultipleCoilsAsync(byte address, ushort startAddress, byte length), WriteSingleCoil(byte address, ushort startAddress, bool value), WriteMultipleCoils(byte address, ushort startAddress, params bool[] value), Sleep(int ms) 等;

**属性 Params** 中, 多个参数是使用 ',' 分隔, 数组使用 '[]', **一定要注意 ',' 特殊意义**, 十六进制数值以 '0x' 开头结合 Action 示例:

1.<Action TargetObj="Window" Method="LoadItem" Params="0"/> 调用显示对象 Window 的方法 LoadItem 其参数为 0

2.<Action TargetObj="BackgroundPlayer" Method="Open" Params="Link,null"/> 打开媒体文件, 只是打开, 是否会播放取决于属性 Background.OpenAndPlay, 以及文件类型

3.<Action TargetKey="SerialPort" Method="SendBytes" Params="[0x00,0x01,0x02]"/> 调用连接对象 SerialPort 发送字节数据,

4.<Action TargetKey="Network.Server" Method="SendMessage" Params="Hello World"/> 调用连接对象 Network.Server, 向所有客户端发送数据

## 【事件 Events】

Player 事件名包括 OnFirstFrame, OnRenderFrame , OnLastFrame, 其中 OnRenderFrame 事件带有 Position 参数值为秒(0.00s),

Button 事件名 Click, 其中 Button 属性为对应按钮集合中的名称

### 【网络接口】

支持方法/属性配置, 示例:

1.向应用服务端口发送: <Action TargetObj="Window" Method="LoadItem" Params="0"/> 会加载 ID 为 0 的 Item

2. 向 应 用 服 务 端 口 发 送 : <Action TargetObj="Window" Method="CallButtonEvent" Params="0,BackgroundButtons,btn\_1"/> 会执行 ID 为 0, BackgroundButtons 中名为 btn\_1 的一组动作集合, 应用场景灵活, 可配置一组虚拟指令专用外部调用

3.向应用服务端口发送: <Window Left="10" Top="10" Width="500" Height="600" /> 配置显示对象的相关属性