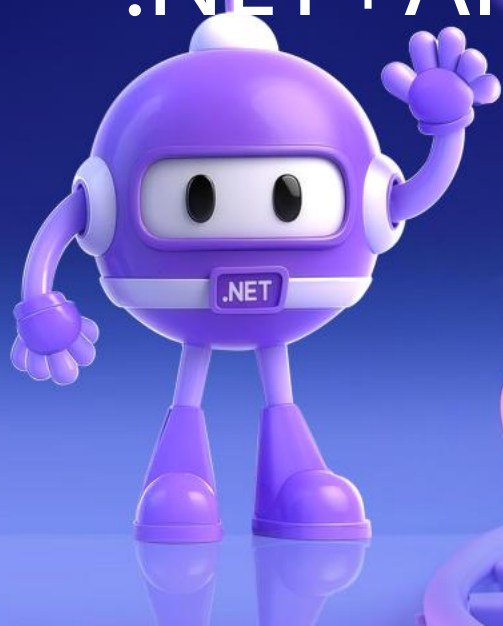


# .NET Conf China 2025

改变世界 改变自己

2025 年 11 月 30 日 | 中国 上海

.NET+AIoT | 打造一个专属陪伴型机器人



.NET Conf China 2025

改变世界 改变自己

# .NET + AIoT 打造一个陪伴型机器人



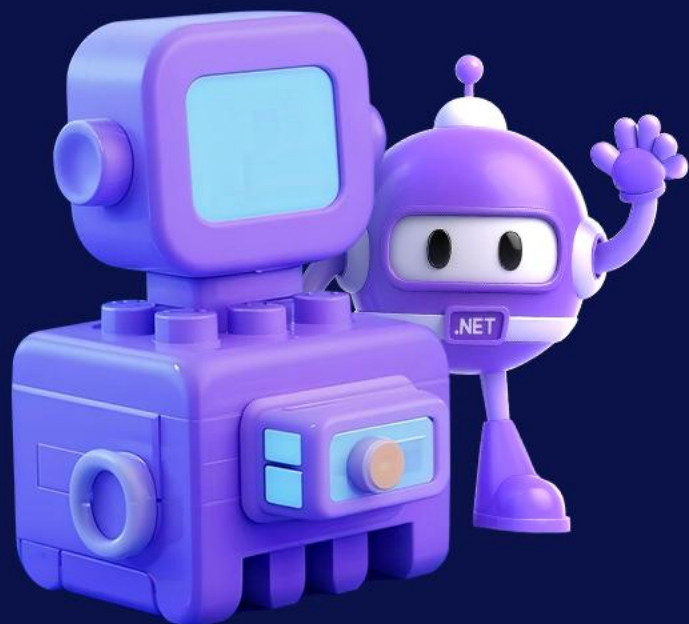
颜圣杰

@sheng-jie



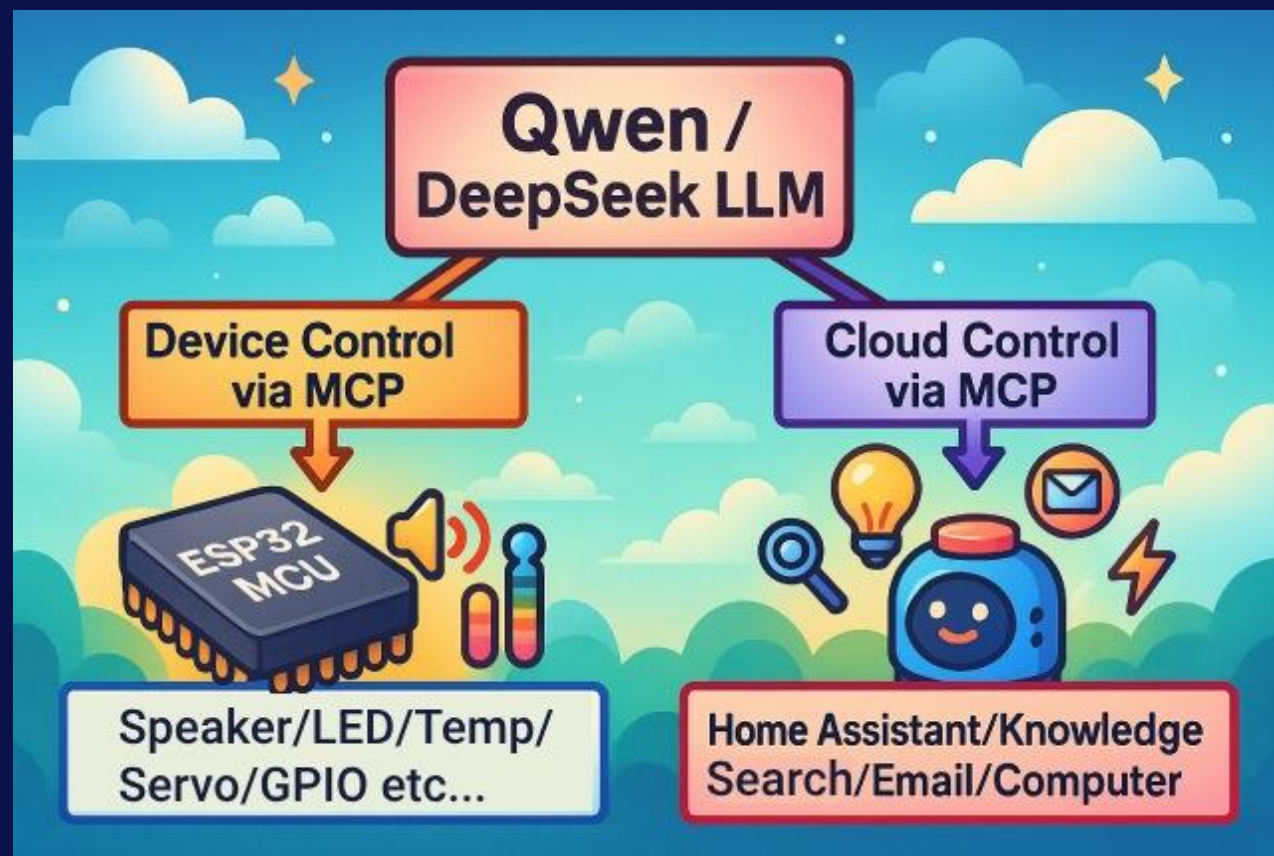
# 开源硬件底座：小智 esp32

小智 esp32，通过 MCP 协议实现多端控制。



.NET Conf China 2025

改变世界 改变自己



<https://github.com/78/xiaozhi-esp32>



# 开源硬件底座意味着什么？

不需要写 C++ 底层  
驱动

把它当作一个标准的  
**WebSocket 客户端**

专注于 .NET 服务端  
逻辑

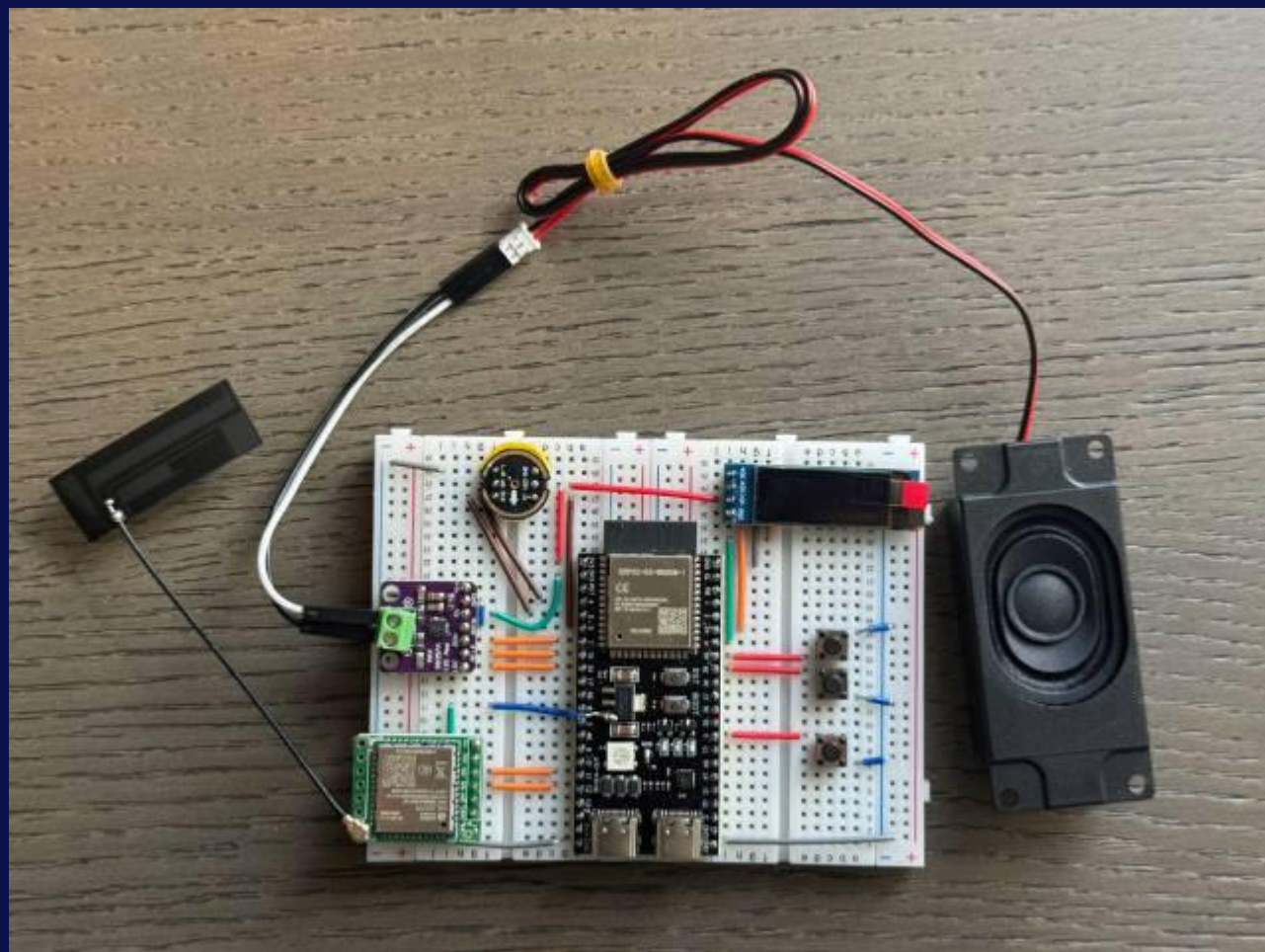
# 认识小智-ESP32

## 核心组件

- 大脑：ESP32

- 听觉：麦克风阵列

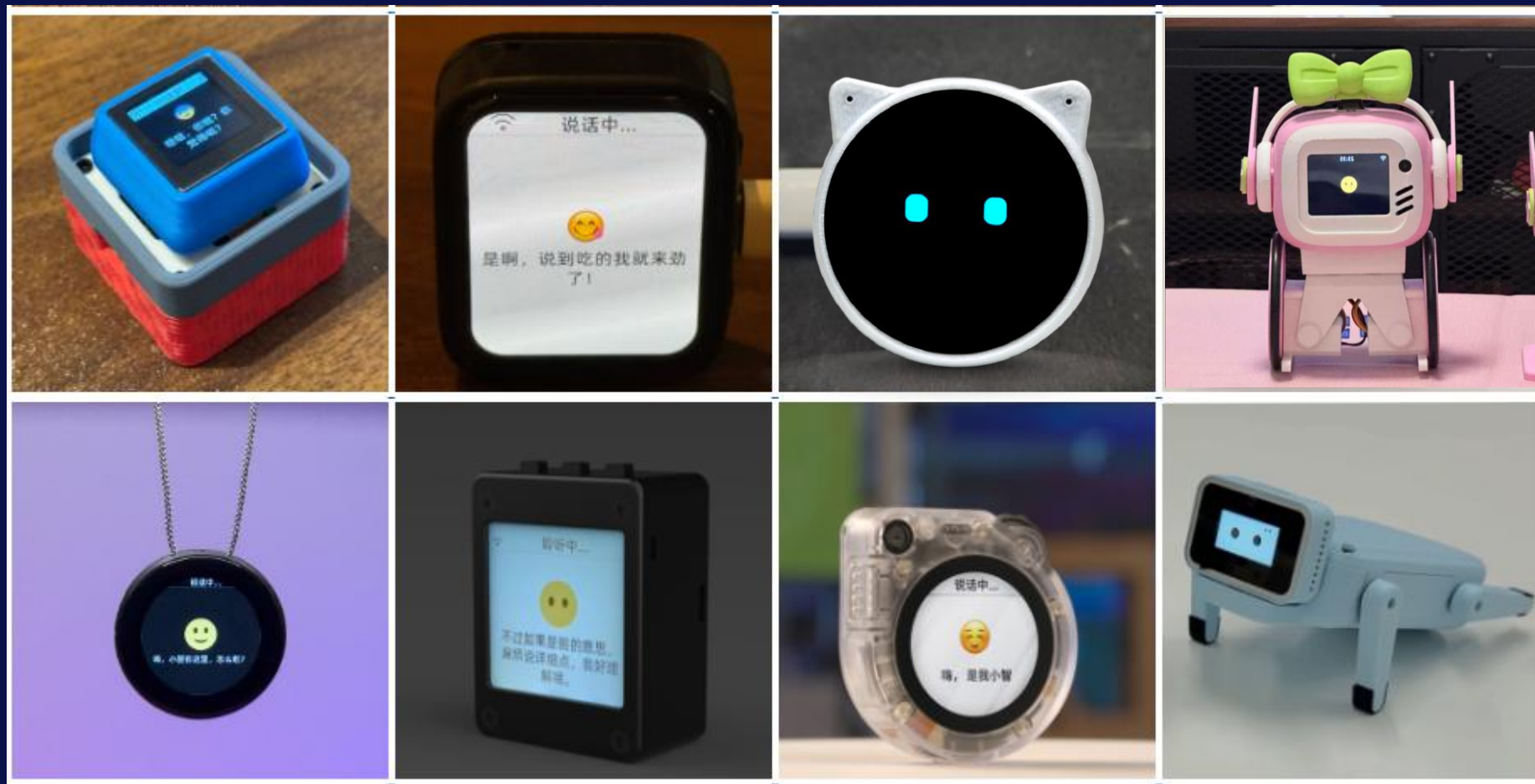
- 表达：扬声器+显示屏



# 小智形态

.NET Conf China 2025

改变世界 改变自己



# 核心功能亮点

.NET Conf China 2025

改变世界 改变自己



## 网络连接

支持 Wi-Fi 与 ML307 Cat.1 4G 模组，随时随地保持在线。



## 多芯片平台支持

适配 ESP32-C3、ESP32-S3、ESP32-P4 系列芯片。



## 全链路语音交互 AI

基于流式 ASR + LLM + TTS 架构，响应迅速，对话自然。



## 离线语音唤醒

集成 ESP-SR 算法，支持低功耗离线关键词唤醒。



## 声纹识别 (3D Speaker)

智能识别当前说话人身份，提供个性化服务。



## OPUS 音频编解码

高效音频压缩，保证通话质量的同时降低带宽占用。



## 双通信协议

灵活支持 Websocket 或 MQTT+UDP 协议栈。



## 交互显示屏

支持 OLED / LCD 屏幕，实时显示丰富表情与状态。



## 设备端 MCP 控制 MCP

直接控制音量、灯光、电机及 GPIO 外设。



## 云端 MCP 扩展 AGENT

扩展大模型能力：智能家居、PC操作、搜索、邮件等。



## 多语言支持

原生支持中文、英文、日文三种语言交互。



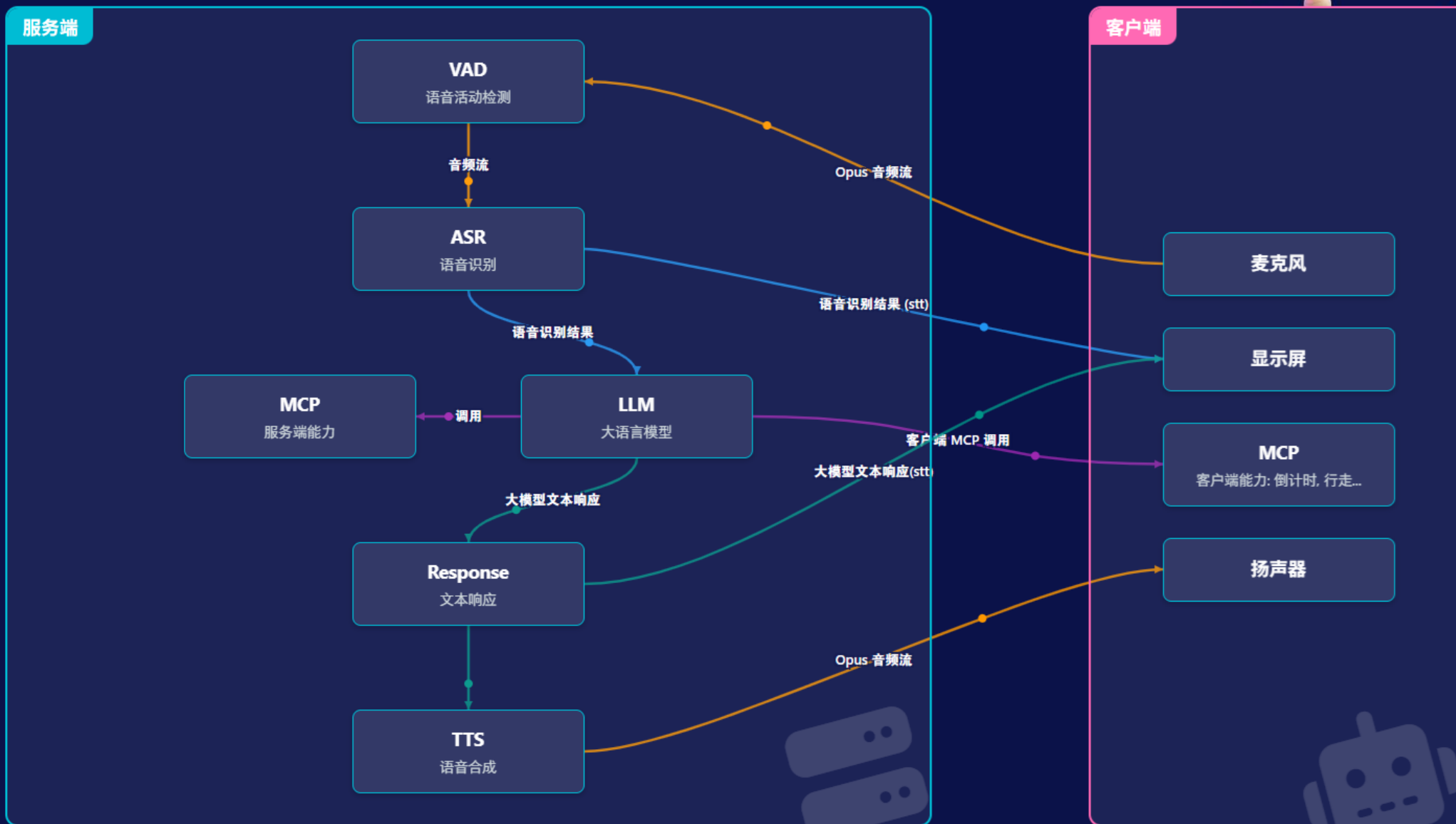
## 电源管理

精准电量显示与高效电源管理系统。

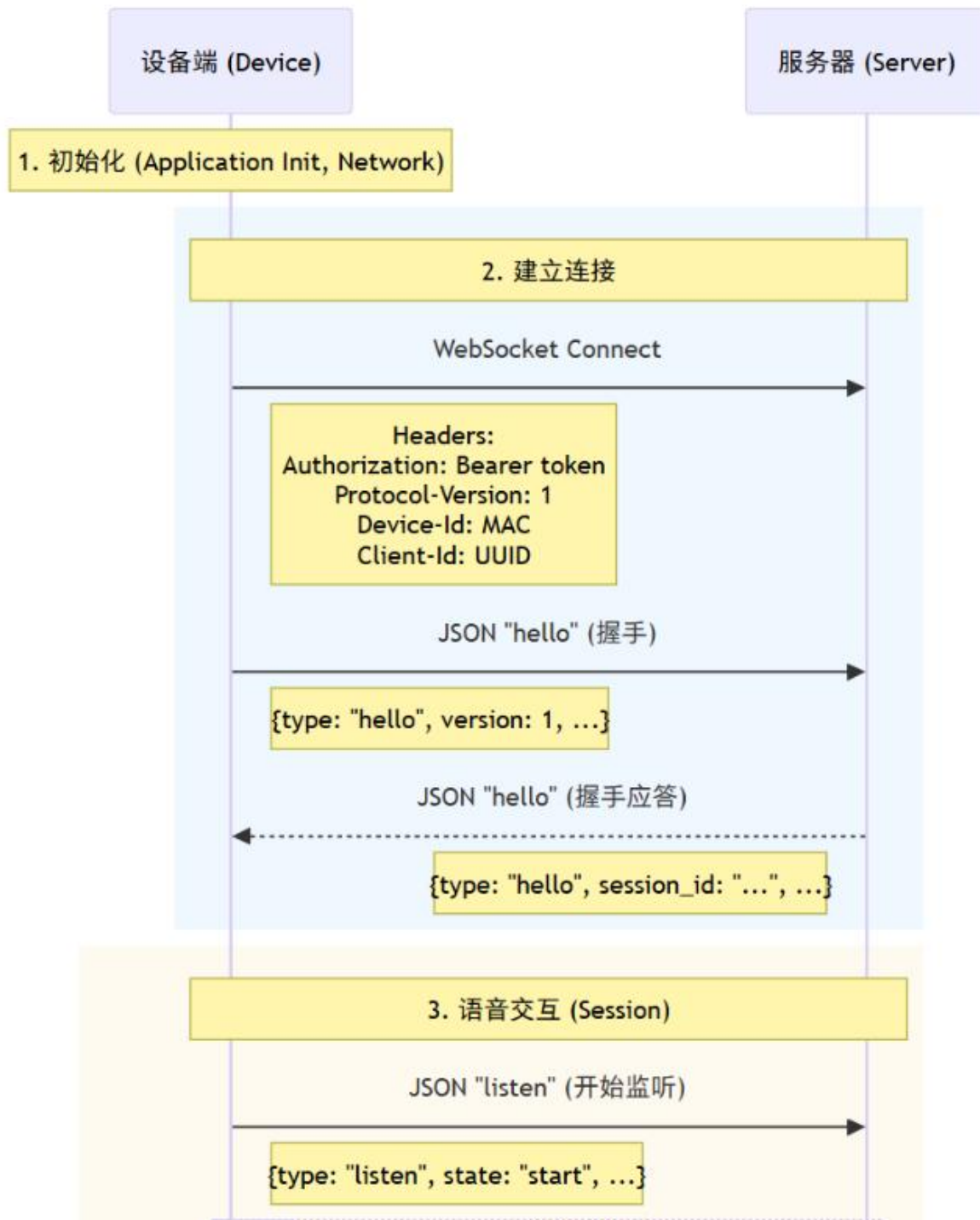
# 交互流程

.NET Conf China 2025

改变世界 改变自己



# 通信协议



.NET Conf China 2025



改变世界 改变自己

握手阶段

# 通信协议



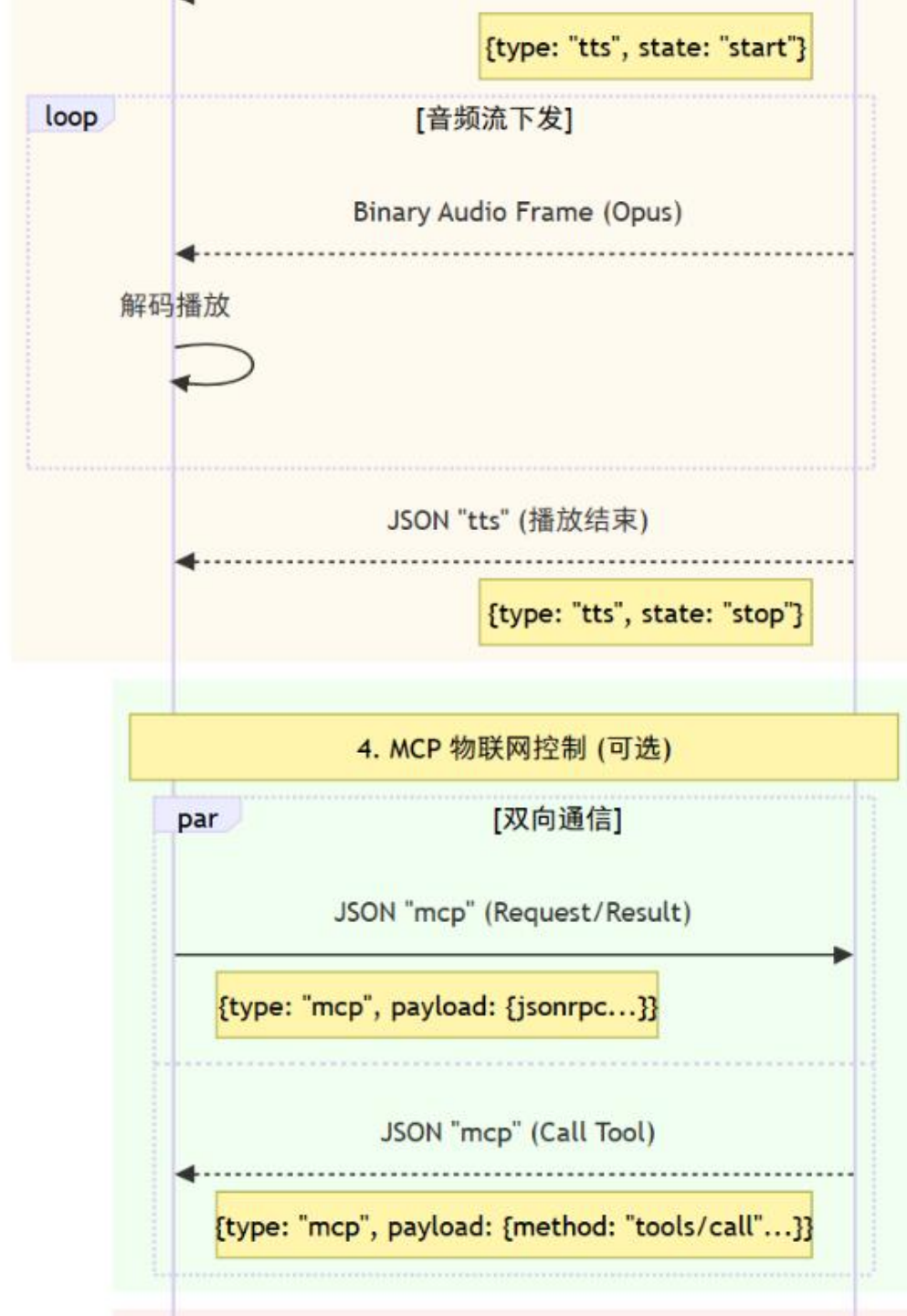
.NET Conf China 2025



改变世界 改变自己

## 语音交互

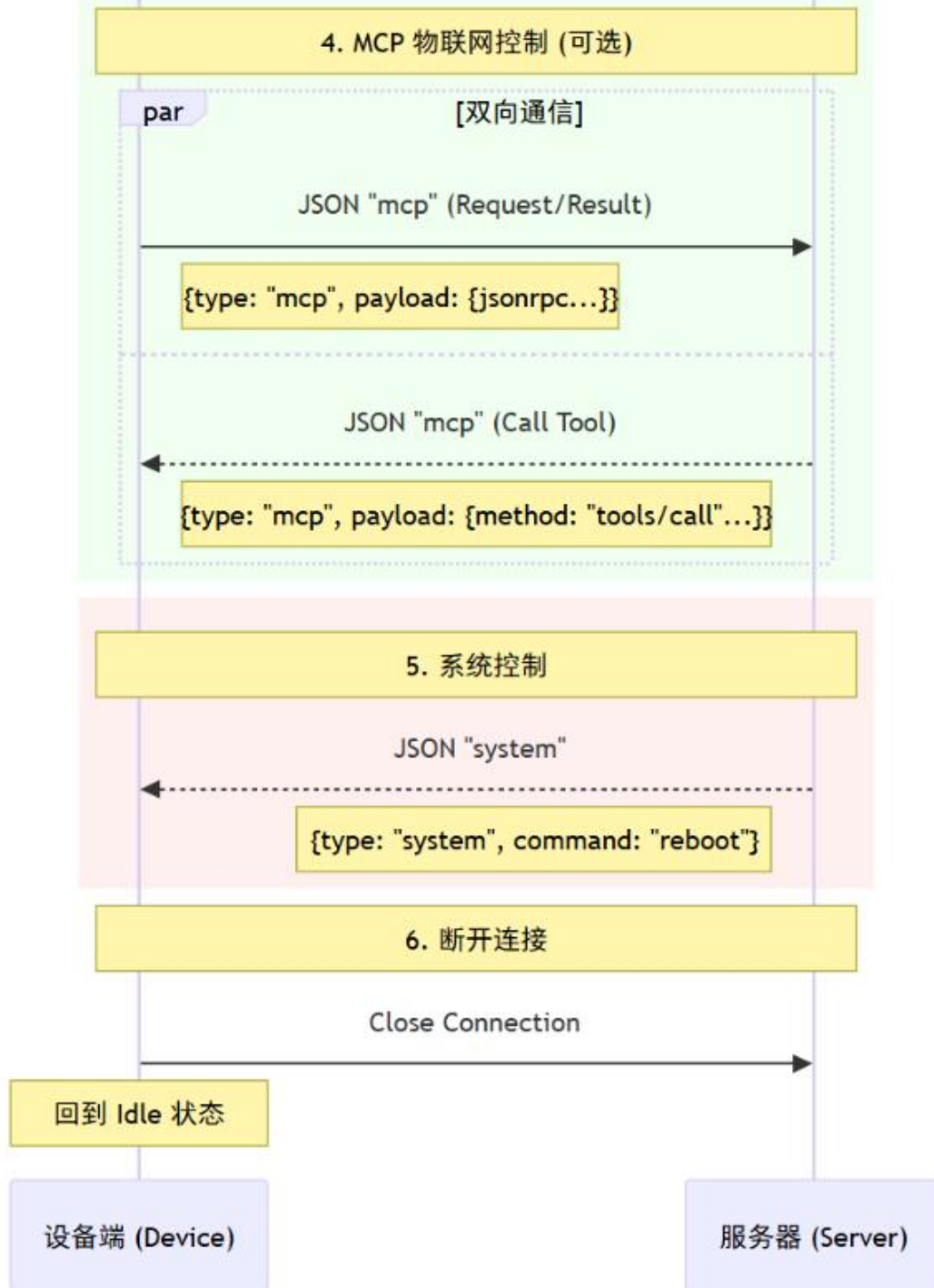
# 通信协议



**.NET Conf China 2025**  
改变世界 改变自己

语音交互

# 通信协议



**.NET Conf China 2025**  
改变世界 改变自己

控制阶段

# 赋予感官 – 听与说 (ASR & TTS)



**听：**音频流实时传输 -> ASR 服务-> 文本

- VAD (语音活动检测) : SileroVAD
- Voiceprint (声纹识别) : 3D-Speaker
- ASR (语音识别) : 火山引擎、科大讯飞、阿里百炼等

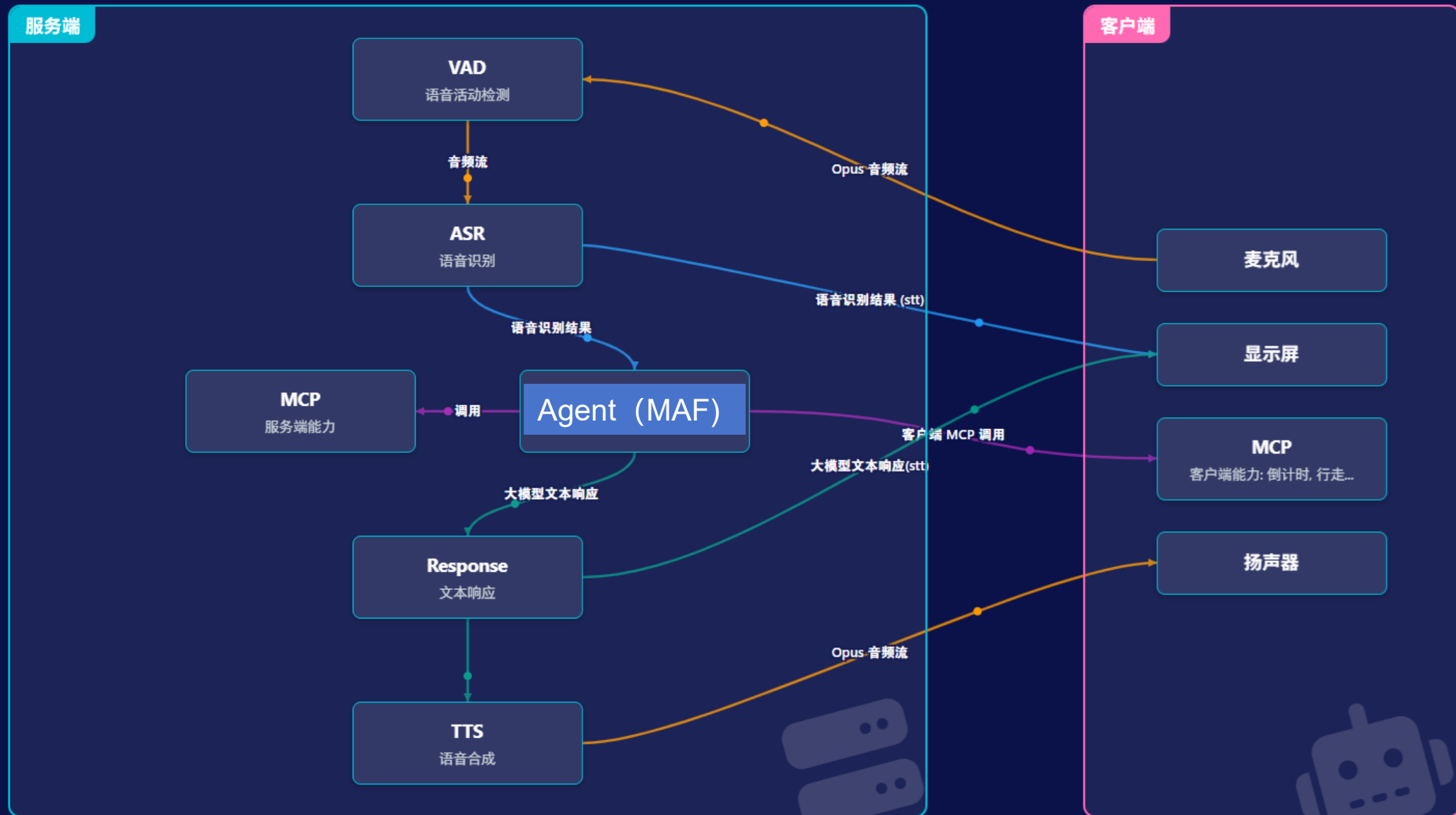


**说：**文本响应 -> TTS 服务 -> 音频流 -> 播放

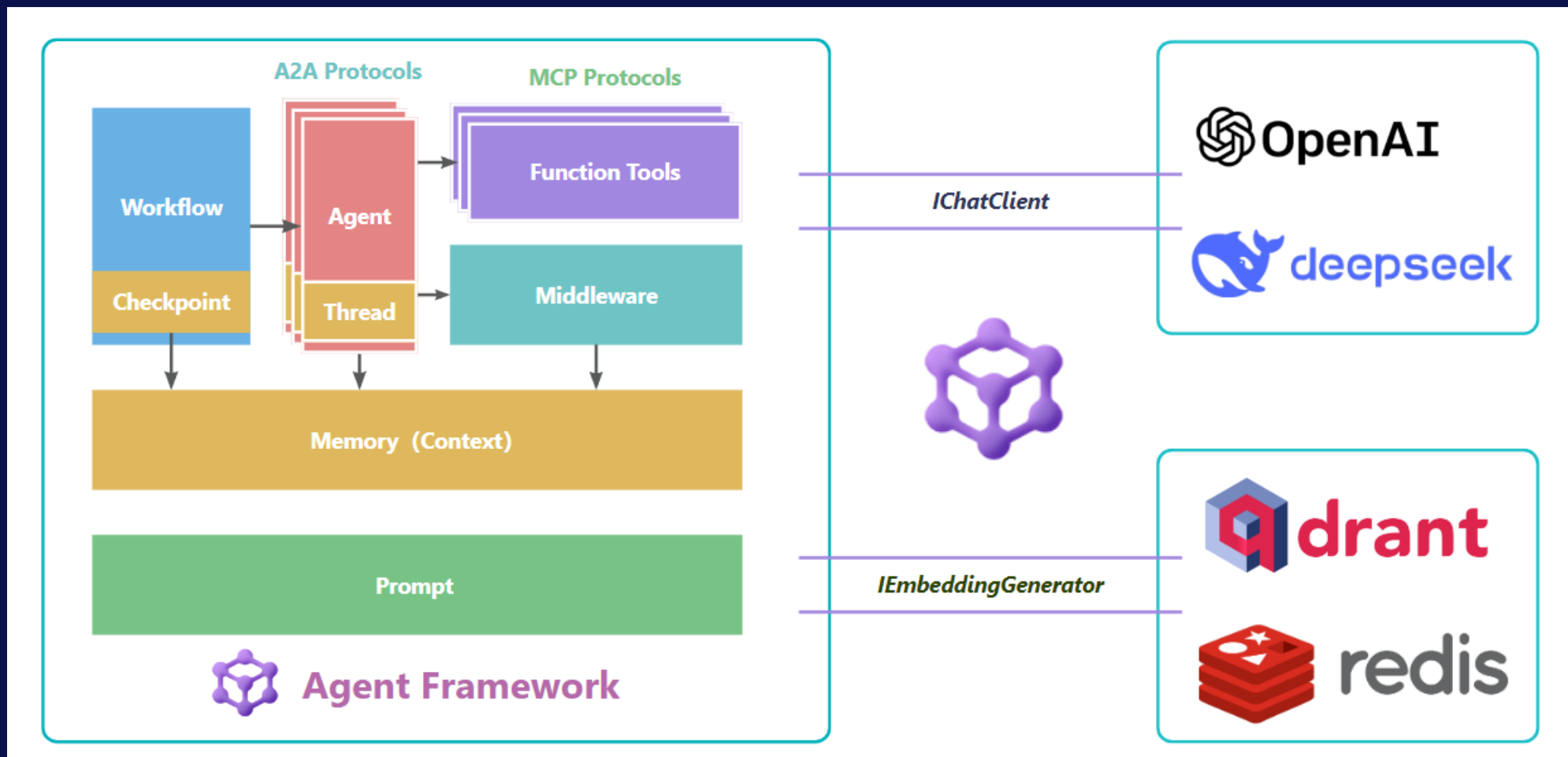
- TTS (语音合成) : EdgeTTS、科大讯飞、火山引擎、腾讯云、阿里百炼等



# 注入灵魂 – 小智大脑 (MAF)



# 注入灵魂 - 智能体大脑 (MAF)



# 基于MAF构建小智Server

通过  
Workflow 构  
建语音交互流  
程

通过 Agent 打  
造智能体

通过 MCP 拓  
展服务端能力

通过 A2A 实现  
多智能体协作

```
var builder = WebApplication.CreateBuilder(args);
var chatClient = AIClientHelper.GetDefaultChatClient();
builder.Services.AddChatClient(chatClient);

builder.AddWorkflow("xiaozi-workflow", (sp, key) =>
{
    var vadExecutor = new VadExecutor(); // Voice Activity Detection
    var asrExecutor = new AsrExecutor(); // Automatic Speech Recognition

    var xiaoziAgent = new XiaoZhiAgent(); // XiaoZhi AI Agent
    var xiaoziExecutor = new XiaoZhiExecutor(xiaoziAgent); // XiaoZhi Executor

    var ttsExecutor = new TtsExecutor(); // Text-to-Speech

    // Build the workflow
    WorkflowBuilder xiaoziWorkflowBuilder = new WorkflowBuilder(vadExecutor);
    xiaoziWorkflowBuilder.AddEdge<VadDetectionResult>(source: vadExecutor, target: asrExecutor,
        condition: vadResult => vadResult?.HasVoice == true);
    xiaoziWorkflowBuilder.AddEdge(source: asrExecutor, target: xiaoziExecutor);
    xiaoziWorkflowBuilder.AddEdge(source: xiaoziExecutor, target: ttsExecutor);

    xiaoziWorkflowBuilder.WithName("xiaozi-workflow");
    var workflow = xiaoziWorkflowBuilder.Build();
    return workflow;
});

var app = builder.Build();

var myworkflow = app.Services.GetKeyedService<Workflow>("xiaozi-workflow");

var audioData = new byte[] { 0x01, 0x02, 0x03, 0x04 };

await using Run run = await InProcessExecution.RunAsync(myworkflow!, audioData);
foreach (WorkflowEvent evt in run.NewEvents)
{
    if (evt is ExecutorCompletedEvent executorComplete)
    {
        Console.WriteLine($"{executorComplete.ExecutorId}: {executorComplete.Data}");
    }
}
```

# 其他资源

- 小智官方服务: <https://www.xiaozhi.me>
- Python 版本小智服务端: <https://github.com/xinnan-tech/xiaozhi-esp32-server>
- .NET 小智服务端: <https://github.com/mm7h/XiaoZhi.Net>
- .NET 小智客户端: <https://github.com/zhulige/xiaozhi-sharp>
- .NET 小智MCP平台: <https://xiaozhi.verdure-hiro.cn>



群聊: .NET+AIoT 社区群



该二维码7天内(11月27日前)有效, 重新进入将更新

**.NET Conf China 2025**

改变世界 改变自己



# THANK YOU