

Amazon 智能音箱 AVS Device

SDK 架构详解（智能音箱的通用架构）

Phodal Huang

September 8, 2017

玩点什么: <https://www.wandianshenme.com>

原文链接:<https://www.wandianshenme.com/play/avs-device-sdk-architecture-overview>

AVS Device SDK 提供基于 C++ (11 或更高版本) 的库, 利用 AVS API 为 Alexa 启用的产品创建设备软件。它是模块化和抽象的, 提供用于处理离散功能 (如语音捕获, 音频处理和通信) 的组件, 每个组件都会显示可以使用和定制的 API, 用于集成。它还包括一个示例应用程序, 演示与 AVS 的互动。

题图说明了构成用于 C++ 的 AVS Device SDK 的组件之间的数据流。

音频信号处理 (**Audio Signal Processor, ASP**) - 将信号处理算法应用于输入和输出音频通道的第三方软件。所应用的算法被设计用于产生干净的音频数据, 包括但不限于声学回声消除 (AEC), 波束形成 (固定或自适应), 语音活动检测 (VAD) 和动态范围压缩 (DRC)。如果存在多麦克风阵列, 则 ASP 构建并输出阵列的单个音频流。

共享数据流 (**Shared Data Stream, SDS**), 一个单生产者, 多消费者的缓冲区, 允许在单个作者和一个或多个读者之间传输任何类型的数据。共享数据流 (SDS) 执行两个关键任务:

- 在发送到 AVS 之前, 在音频前端 (或音频信号处理器)、唤醒语引擎和 Alexa 通信库 (ACL) 之间传递音频数据
- 通过 Alexa 通信库将由 AVS 发送的数据附件, 传递给特定能力的代理

共享数据流在产品特定的内存段 (或用户指定的) 上的环形缓冲区的顶部实现, 允许将其用于进程内或进程间通信。请记住, 作者和读者 (们) 可能处于不同的线程或进程中。

唤醒词 (**Wake Word Engine, WWE**), 是一个在输入流中识别唤醒词的软件。它由两个二进制接口组成: 第一个进行唤醒字识别 (或检测), 第二个处理特定的唤醒词模型 (在这种情况下是 "Alexa")。这取决于你的实现, WWE 可以在片上系统 (SOC) 或专用芯片上运行, 如数字信号处理器 (DSP)。

音频输入处理器 (**Audio Input Processor, AIP**), 用于处理通过 ACL 发送到 AVS 的音频输入。这些包括设备上的麦克风, 远程麦克风和其他音频输入源。

AIP 还包括了在不同音频输入源之间切换的逻辑。在给定的时间里, 只能将一个音频输入源发送到 AVS。

Alexa 通讯库 (**Alexa Communications Library, ACL**), 作为客户端与 AVS 之间的主要通信渠道。ACL 执行两个主要任务:

- 建立和维护与 AVS 的长期持续连接。ACL 遵守消息规范的详细说明, 见: [Managing an HTTP/2 Connection with AVS](#)

- 提供消息发送和接收功能，其中包括支持 JSON 格式的文本和二进制音频内容。
详见：[Structuring an HTTP/2 Request to AVS](#)

Alexa 指令排序器库 (Alexa Directive Sequencer Library, ADSL)，管理 AVS 指令的顺序和顺序，详见：[AVS Interaction Model](#)。该组件管理每个指令的生命周期，并通知指令处理程序（可能是也可能不是能力代理 (Capability Agent)）来处理消息。

活动焦点管理库 (Activity Focus Manager Library, AFML)，为设备提供视听焦点的集中管理。焦点是基于频道的，详见：[AVS Interaction Model](#)，用于管理视听输入和输出的优先级。

频道可以在前台或后台。在任何给定的时间，只有一个通道可以在前台和焦点。如果多个通道处于活动状态，则需要遵循以下优先顺序：对话框 (Dialog) > 警报 (Alert) > 内容 (Content)。当在前台的频道变为不活动时，优先顺序中的下一个活动通道将进入前台。

焦点管理并不限于能力代理 (Capability Agents) 或指令处理程序，也可以由非 Alexa 相关代理使用。它允许所有使用 AFML 的代理在设备上保持一致的焦点。

能力代理 (**Capability Agents**)：处理 Alexa 驱动的交互；具体指示和事件。每个能力代理对应于由 AVS API 公开的特定接口。这些接口包括：

- [SpeechRecognizer](#) - 语音捕获接口。
- [SpeechSynthesizer](#) - Alexa 语音输出接口。
- [Alert](#) - 用于设置、停止和删除定时器和警报的界面。
- [AudioPlayer](#) - 用于管理和控制音频播放的界面。
- [Notifications](#) - 显示通知指示器的界面。
- [PlaybackController](#) - 用于通过 GUI 或按钮导航播放队列的界面。
- [Speaker](#) - 音量控制界面，包括静音和取消静音。
- [System](#) - 用于将产品状态/标志传达给 AVS 的接口。
- [TemplateRuntime](#) - 用于呈现可视化元数据的界面。

原文链接：<https://github.com/alexavs/avs-device-sdk>

原文链接：<https://www.wandianshenme.com/play/avs-device-sdk-architecture-overview>