Open Voice OS

Presented By: Aditya Mehra

What is Open Voice OS

- Open Voice OS or OVOS as we like to call it in short aims to be a full operating voice assistant stack that is free and open source, the stack can be better defined as:

- Stack as a Framework

- Stack as an Operating System

- Before diving deeper into the above.. let's see what its built upon.

How did OVOS start ?

- OpenVoiceOS originally started as heavily patched Mycroft Stack running on an embedded buildroot system for DIY Raspberry Pi smartspeakers

- Mycroft community developers interested in supporting more platforms and expanding the stack

- Not under the control of any one organization

- Community First!

The internals

- OVOS consist of python based core framework called ovos-core, this core framework is responsible for providing distinct services

- Some important services it provides are:
    - Websockets Messagebus Service
    - Skills Service
    - Audio Service
    - Speech Service
    - GUI Service

The internals – Websocket Messagebus

- The websocket messagebus service is our master communication layer, everything in the OVOS world talks to each other on this layer

  - Service A

```python
self.bus.emit(Message("this.is.an.example", data={"foo": "bar"}))
```
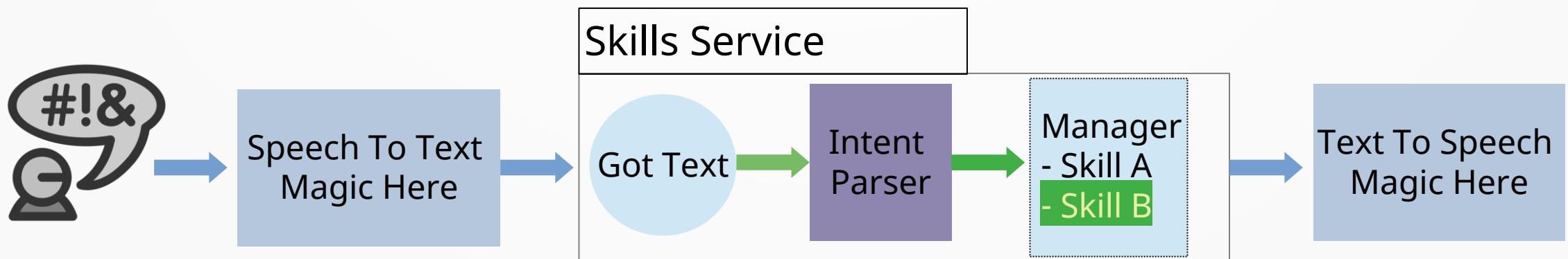
  - Service B

```python
        self.bus.on("this.is.an.example", self.do_something)

def do_something(self, message):
    msg = message.data.get("foo")
    print(msg)
    self.bus.emit(message.reply("this.is.an.example.response", {"bar": "baz"}))
```

The internals – Skills Service

- Mini voice interface apps we call skills live along side the intent parsers here

- Provides a manager who's sole responsibility is lifecycle management of skills

- intent engines parse text to actions here

The internals – Speech Service

- Housing for the Wake Word Engine API

- Housing for the Speech To Text Engine API

- Responsible for everything in the speech input process
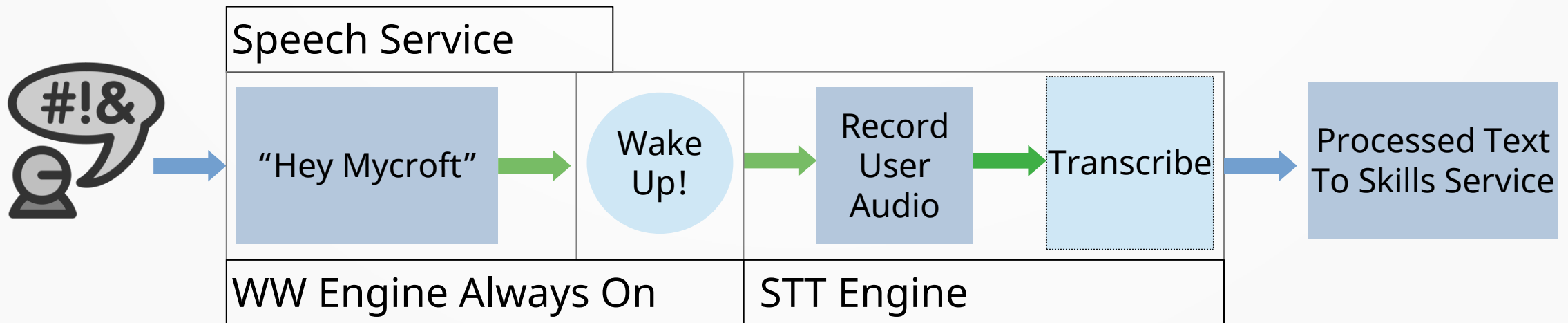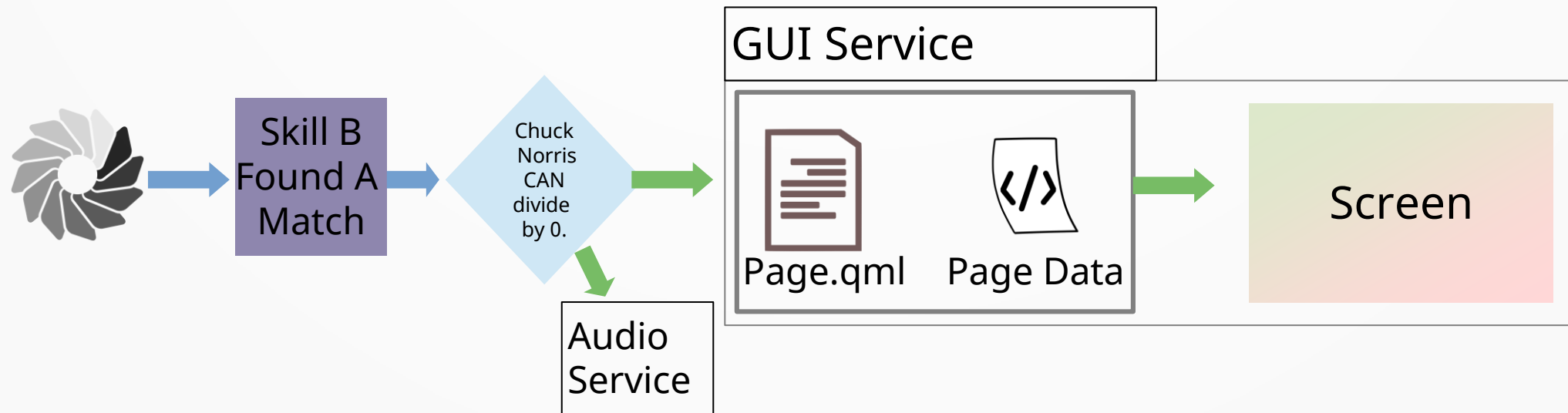
The internals – Audio Service

- Housing for the Text To Speech Engine API

- Housing for the Playback Services / Multimedia Framework

- Responsible for everything in the audio output process
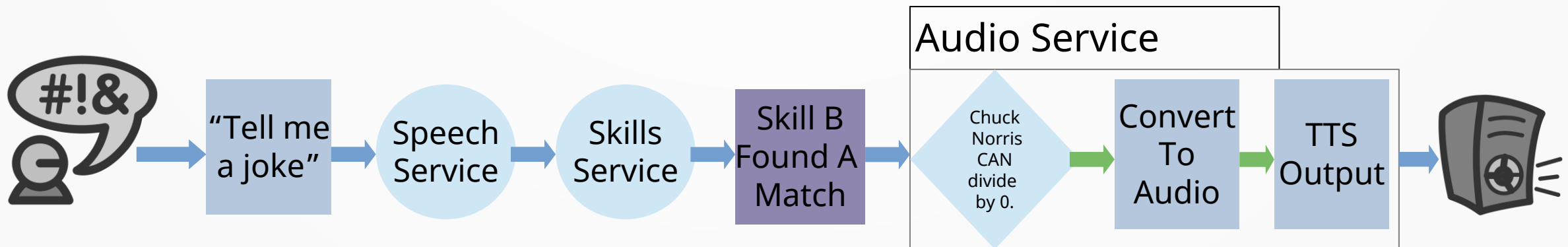
The internals – Gui Service

- Talks to Mycroft Gui in the gui protocol language
- Responsible for skills display lifecycle management and skills display data management

The externals

- Apart from the core framework there other major external frameworks that build up our stack

- OVOS Shell

- OCP Multimedia Framework

OVOS Shell

- Kirigami based application running on EGLFS using the Mycroft GUI protocol view

- A GUI Interface developed for our smartspeaker and embedded platforms

- Uses a custom Kirigami Platform Theme plugin to provide global themeing to all our skills and user interface

- Integrates with KDE Connect out of the box
  - Pairing
  - Playback control via MPRIS

OVOS Shell Demo

Network & Wifi Setup Quick Access

- Lets see some of the things the shell can do

OCP Multimedia Framework

- OVOS Common Play is a full-fledged voice media player framework handling voice integration and playback functionality for multimedia related skills

- OCP handles all queries related to media playback such as "play a song", "play a video", "play anything"

- OCP framework can be extended by multiple media providers and uses fuzzy matching to find the best results to your media query

OCP Demo
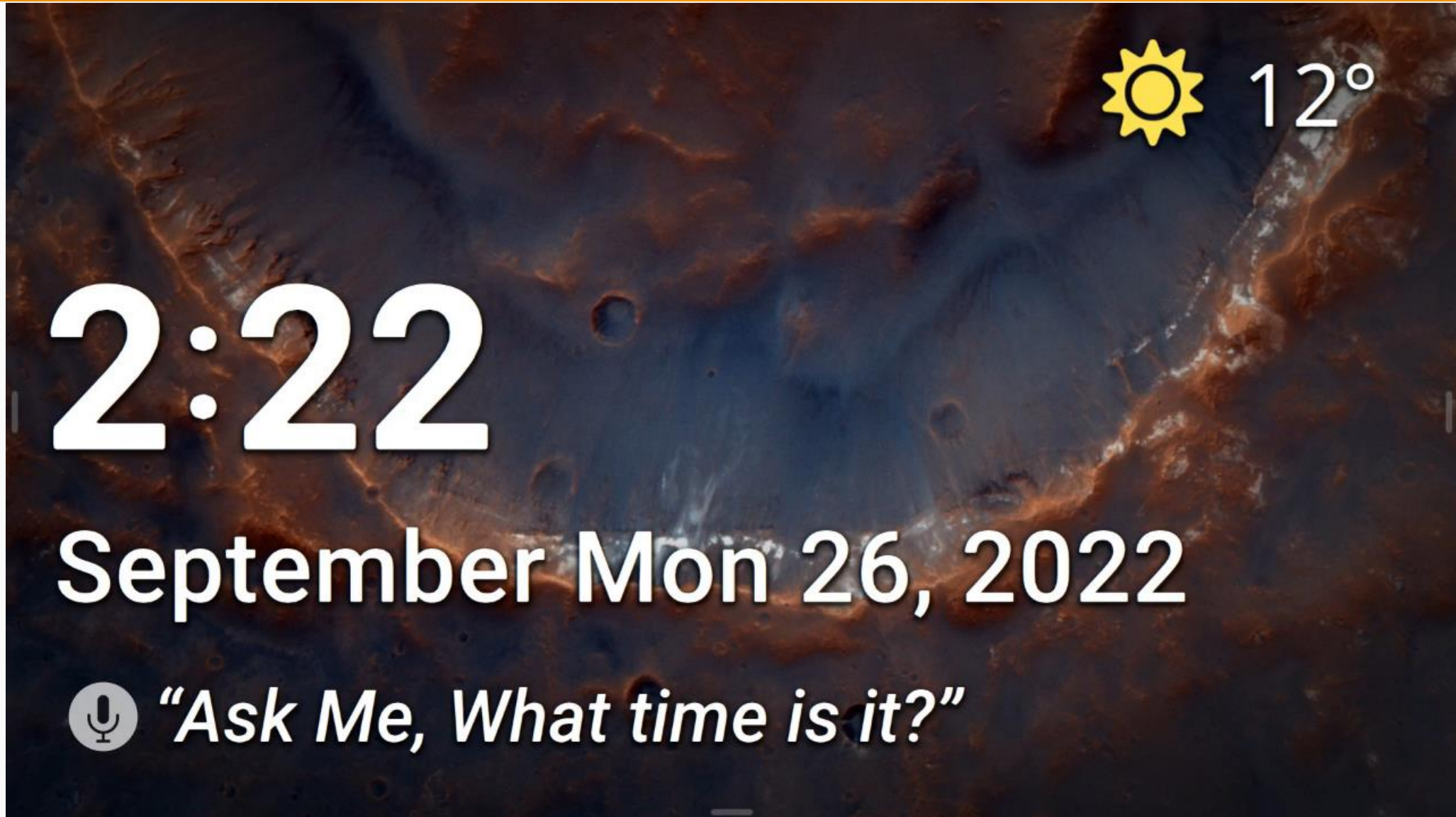
- Lets see the multimedia framework in action

How is ovos different internally

Mycroft Core	OpenVoice Core

Mycroft Core

- Monolithic architecture

- Supports limited stt, tts engines hardcoded in mycroft-core

- Tied up to Mycroft's selene backend

OpenVoice Core

- Micro / plugin based architecture

- Supports multiple STT providers: Google STT, VOSK, Conqui-AI STT, more

- Supports Mimic, Mimic 1, Mimic 2, Mimic 3, Larynx, Glow, Pico, Mary TTS, Conqui-AI TTS engines

- Local/Offline by default, Mycroft's selene backend optional

OVOS Plugin Manager

- OVOS Plugin Manager can be used to search, install, load and create plugins for the OpenVoiceOS ecosystem

- Every engine is a plugin in the ovos world

- Types of plugins supported:

  - Wake Word Engine Plugins
    - Precise, Precise-Lite, Snowboy, Pocketsphinx, Nyumaya, Hotkeys, Vosk...
  - Speech To Text Engine Plugins
    - Selene, Vosk, Pocketsphinx, Google, Self-Hosted HTTPS Service...
  - Text To Speech Engine Plugins
    - Mimic 1/2/3, Pico, Polly TTS, Google, Mary TTS, Conqui-TTS, Softcala, Cataron, Cotovia, ResponsiveVoice...
  - Intent Engine Plugins
    - Adapt, Padaos, Padatious, Regex..
  - Hardware Abstraction Plugins

Stack as a framework

- OVOS incorporated into your own project

- Utilize a subset of the services and plugins to meet the product requirements

- Connect up your native QT Applications to service a voice interface using the libqmycroft library

Stack as a framework demo
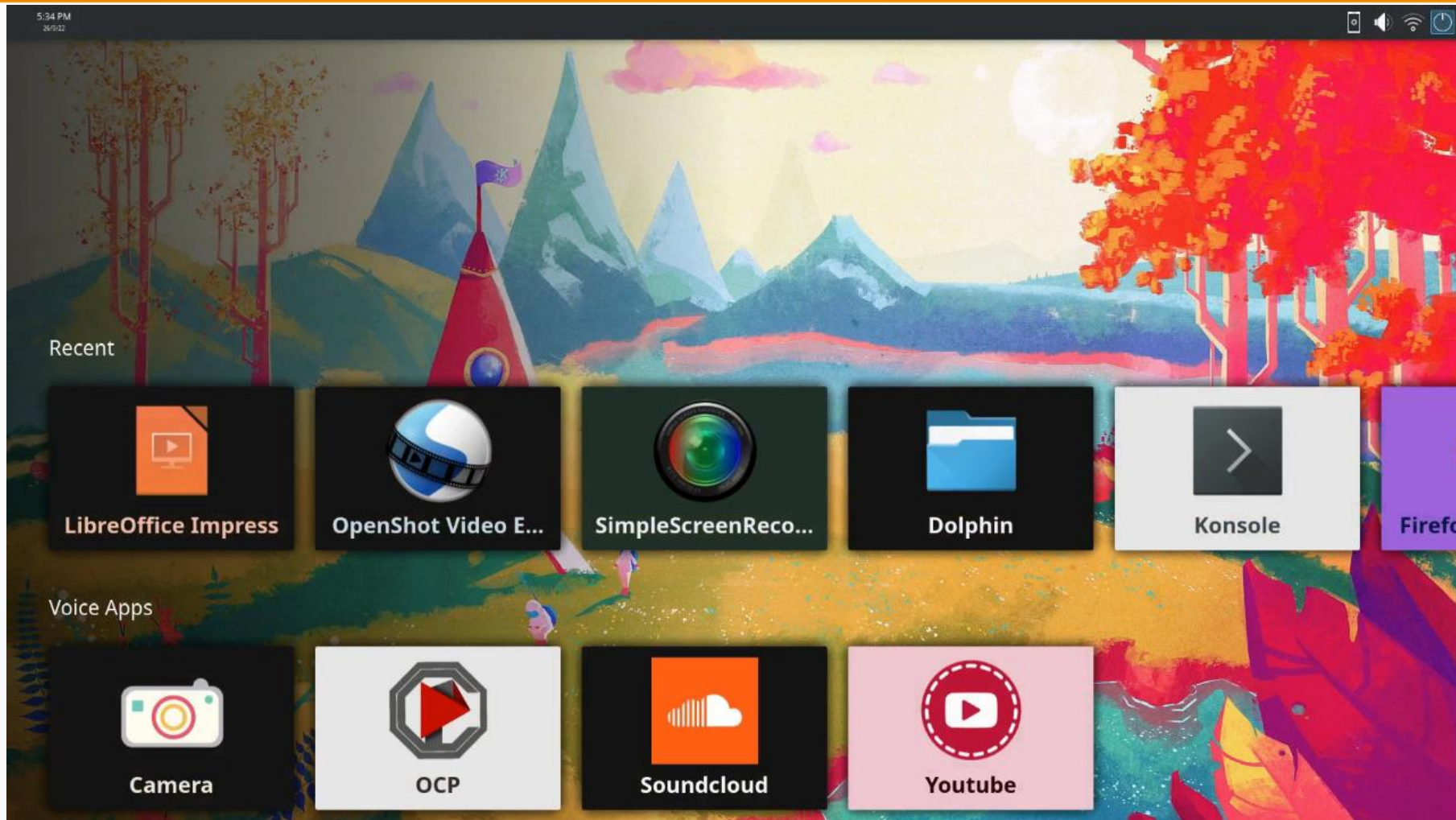
- OVOS integrated into Plasma Bigscreen to serve voice applications and voice queries

Stack as an operating system

- OVOS on bare metal as an embedded system
- Smartspeakers, Voice Satellites, Magic Mirrors
- Utilizes plugin based hardware abstraction layer
  - Mic Array and Audio Board Plugins
  - Network Management Plugins
  - Display Management Plugins
  - GPSD Plugins

# Stack as an operating system demo
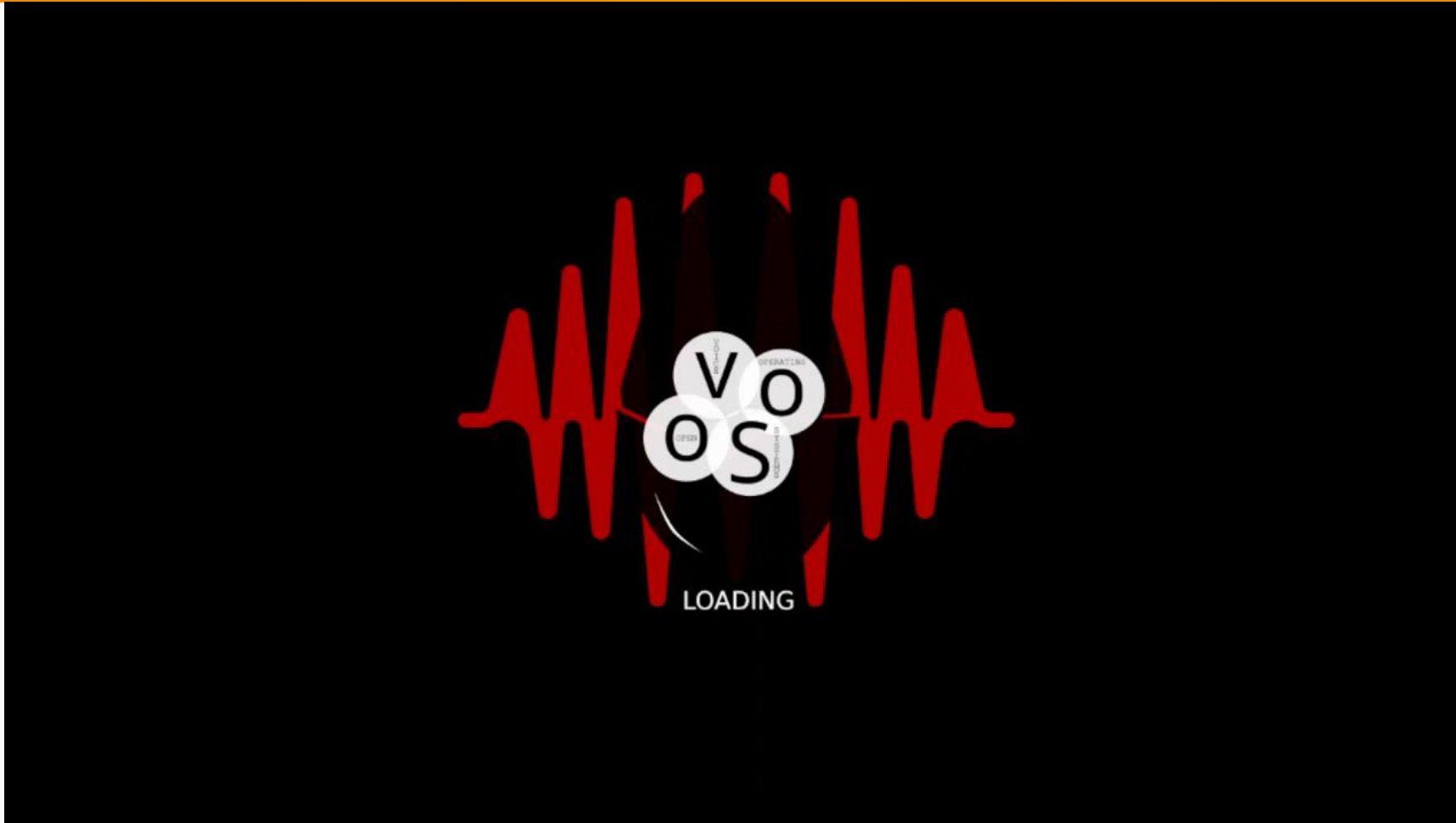
- OVOS running our embedded shell and packages as a smartspeaker

Targeting All Platforms

- Embedded headless device or single board computers with screens to build a do it yourself smart speaker, Openvoice OS is the open community playground for all platforms to come and experiment at

- You can expirement with us on:
  - Buildroot Based Image for Mark-2 Dev Kit
  - Manjaro Arch Based Image for Mark-2 Dev Kit
  - Manjaro Arch Based Image for Respeaker based HW
  - Manjaro Arch Based Plasma Bigscreen Images

Meet Our Team

- Open Voice OS is a community powered project with contributors from various parts of the globe working in various industries

- The project apart from being community powered also consist of lead maintainers for various aspects of the framework

- Lead Maintainers:
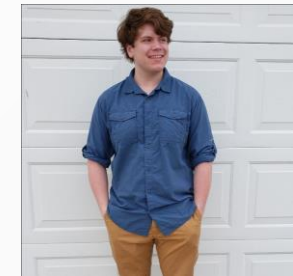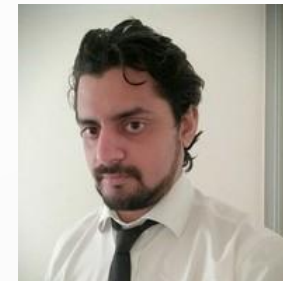
- **Casimiro Ferreira (Jarbas)**

- Community mycroft-core developer since 2017

- Core Maintainer / Dev

- PHAL Maintainer

- **Daniel McKnight**

- **(Daniel)**

- Community mycroft-core developer since 2018

- Core Maintainer / Dev

- **Peter Steenbergen (J1nx)**

- Buildroot and

- OVOS Packaging Maintainer

- Embedded and Audio Hardware Bring Up

- Image Build Services

- **Aditya Mehra (Aix)**

- OVOS Shell and

- OVOS Core GUI Maintainer

- OVOS API Service Maintainer

- Skills UI and more

- **Parker Seaman (Strongthany)**

- Community and Marketing Manager

- Documentation and Developer Relations

Thank You!

Find us on:
https://www.openvoiceos.com
https://github.com/OpenVoiceOS
Matrix channel links on the website :)