



Automated Air-Ground Communication with GStreamer



COLLABORA

Open First



COLLABORA

Open First



COLLABORA

Open First

Urban Air Mobility (UAM)



COLLABORA

Open First

Urban Air Mobility (UAM)

- Highly congested urban environment
- Electric vertical takeoff and landing (EVTOL)
- Semi or fully automated aircraft
- Automate non-critical communication to reduce cognitive load on human ground operators



Urban Air Mobility (UAM)





COLLABORA

Prototype Design

Prototype Design

- GStreamer WebRTC pipeline
- Server 1: automated speech to text (STT)
- Server 2: large language model (LLM)
- Server 3: automated text to speech (TTS)



Prototype Design

- WebRTC simplifies server design
- STT receives audio, transcribes, sends text back
- LLM listens for STT, generates response, sends text back
- TTS listens for LLM, synthesizes, sends audio back



Prototype Design

- Python all the way down
- LiveKit SFU – Swift and Python clients
- `livekitwebbrtcsrc/livekitwebbrtcsink + appsink/appsrc`
- OpenAI Whisper STT (Faster-Whisper)
- Coqui TTS



Prototype Tuning

Prototype Tuning

- Segment analysis via pySiler VAD (voice activity detector)
- Mixture of languages : English and Korean



Prototype Tuning

- Solution: Whisper initial prompt
- Feed a pre-compiled list of phrases into the prompt



Future Plans

- Implement LLM – Mistral, Llama or Phi3
- Reduce latency using Whisper Live streaming
- Replace Coqui with Whisper Speech
- Port to Python Analytics – replace three servers with single GStreamer pipeline



Source Code ?

- Open sourcing currently in progress
- Check <http://www.collabora.com/news-and-blog/?blogs> for details

Appreciation

- Thank you to fellow Collaborans Olivier Crete, Daniel Morin, Marcus Edel, Vineet Suryan and Jordan Yelloz
- Thank you to Justin Kim at SKT





Thank you!



COLLABORA

Open First



We are hiring
col.la/careers



COLLABORA

Open First