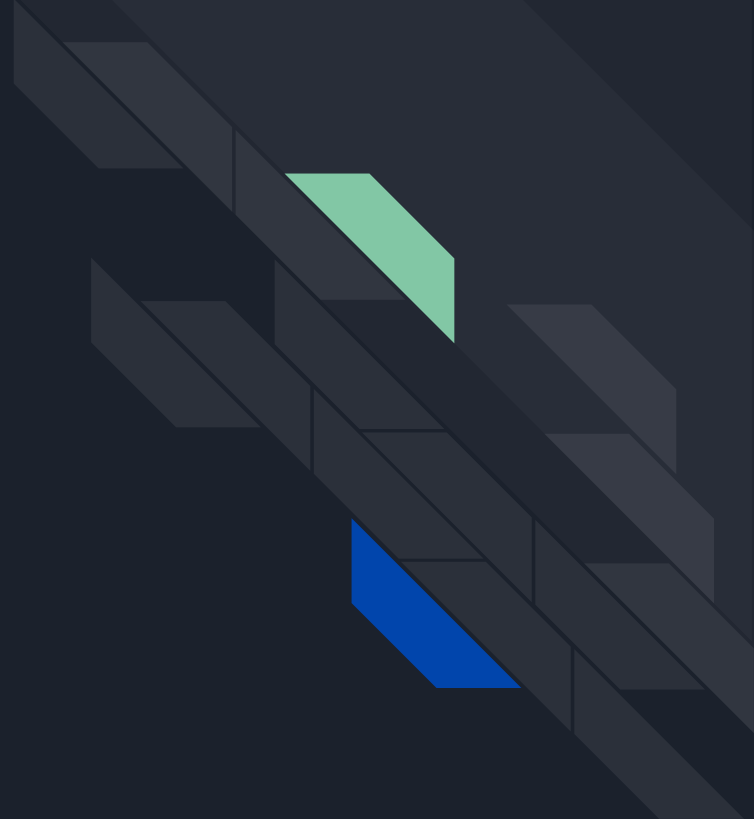# azuracast

Liquidsoap and the Open-Source Web Radio Ecosystem

Buster "Silver Eagle" Neece
Creator and Lead Developer, AzuraCast
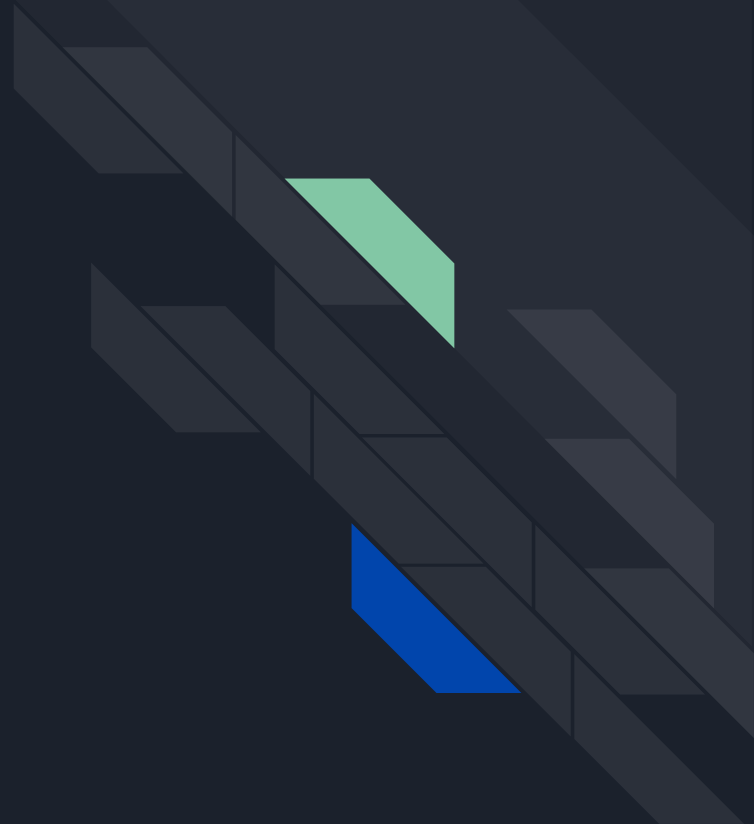
About Me

Buster "Silver Eagle" Neece

Yes, Buster is my real name!

- "Full-stack" web applications developer
- Specialty in PHP on the backend and JS (esp. VueJS) on the frontend
- Member of lots of online fan communities (hence the nickname)
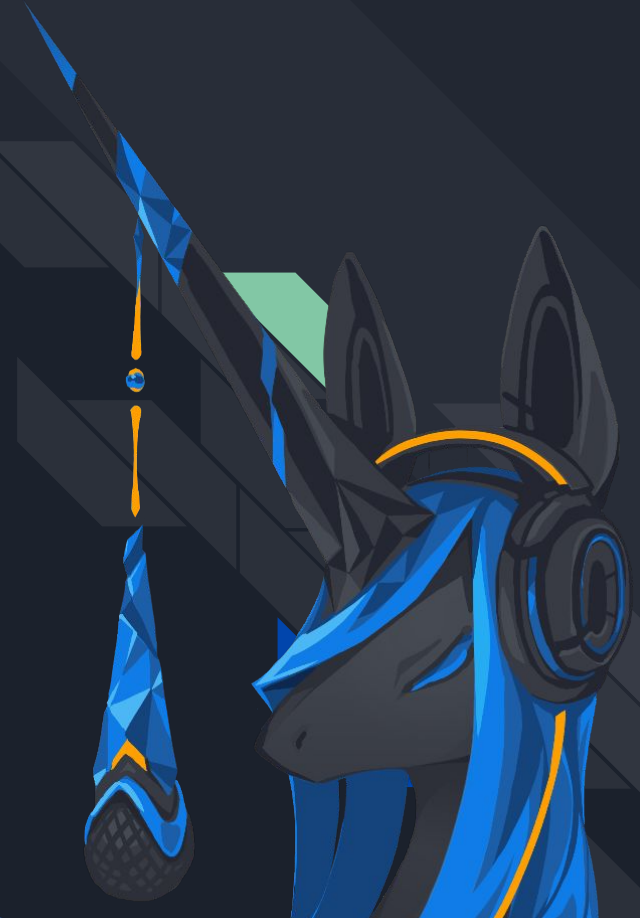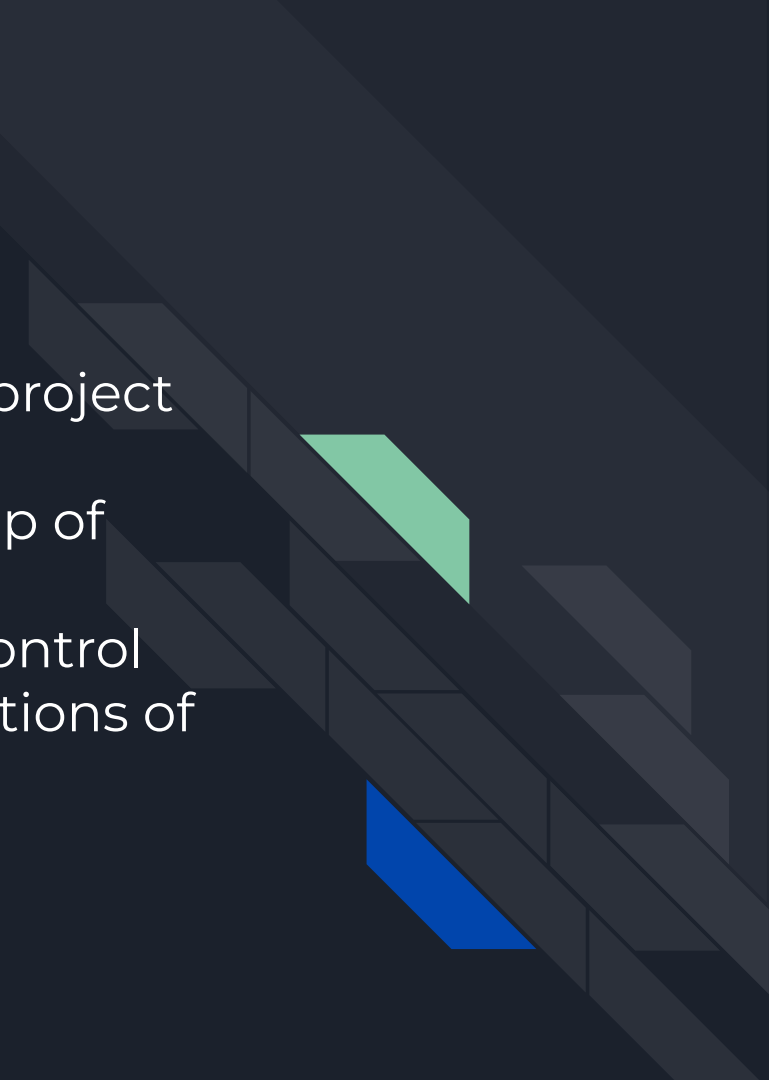
About AzuraCast

AzuraCast is…

- A "turnkey" self-hosted web radio management suite
- An abstraction layer above broadcasting software, AutoDJ software, and DevOps concerns
- Pre-built installers and Docker images
- A web UI to manage it all

Before AzuraCast…

- A fandom-specific open-source project called "Ponyville Live"
- An abstraction layer sitting on top of Centova Cast
- Repurposed in 2016 to directly control broadcasting tools and serve stations of any size
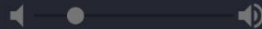
AzuraTest Radio

Flux
Sebastian Kamae, Dayle

00:02 ━━━━━━━━━━━━━━━━━━━━━━━━━━━ 02:49

▶ **/RADIO.MP3 (128KBPS MP3)** ▼   🔈 ●━━━━━ 🔊

⟳ SONG HISTORY   ❓ REQUEST SONG   ⬇ PLAYLIST

azuracast local development

AzuraTest Radio
11:33:47 AM UTC

- 🖼 Profile
- 🌐 Public Page ↗
- 🎵 Music Files
- 🎵 Playlists
- 🎙 Podcasts
- 🎙 Streamer/DJ Accounts
- 📻 Web DJ ↗
- 📡 Mount Points
- 📶 Remote Relays
- ⟨⟩ Web Hooks
- 📋 Reports
- ⚙ Utilities

AzuraTest Radio
A test radio station.

✏ **EDIT PROFILE**

On the Air

🎧 **0 Listeners**
0 Unique

♪ **Now Playing**

Cascada
Juan Rios
Playlist: default
01:51 / 03:06

▶ **Playing Next**

Flux
Sebastian Kamae, Dayle
Playlist: default

▶ **SKIP SONG**

Scheduled

Playlist
Test Schedule Thing

10:00 AM - 10:00 PM
Now

Playlist
Test Schedule Thing

Jan 21, 2022, 10:00 AM - 10:00 PM
in 22 hours

Song Requests Enabled

Streamers/DJs Enabled

Streams

Local Streams	Listeners
▶ /radio.mp3 (128kbps MP3) https://azuracast.local/radio/8000/radio.mp3	🎧 0 0 Unique
▶ /mobile.mp3 (64kbps MP3) https://azuracast.local/radio/8000/mobile.mp3	🎧 0 0 Unique

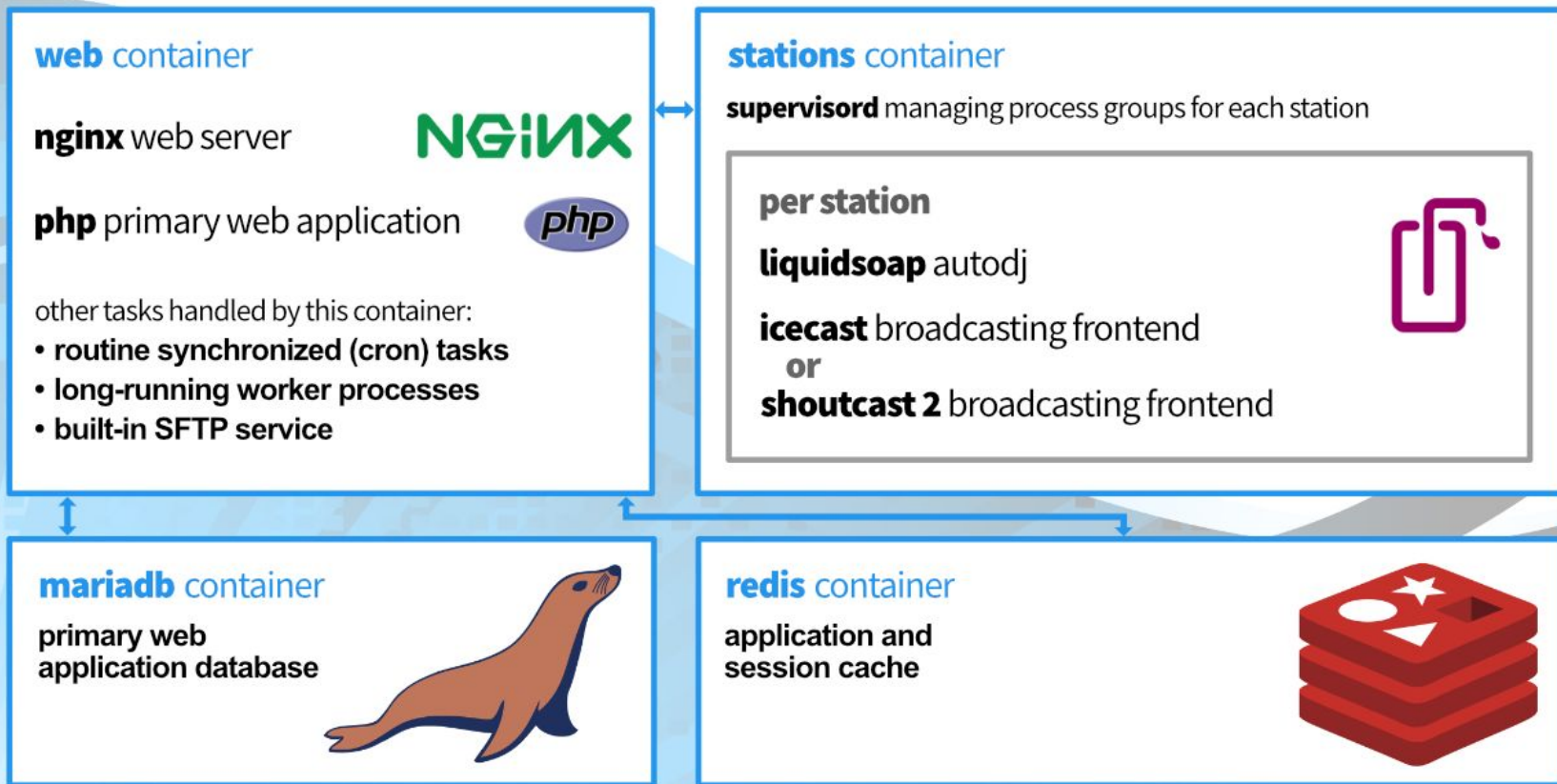⬇ **DOWNLOAD PLS**　⬇ **DOWNLOAD M3U**

Broadcasting Service Running
Icecast

Administration	https://azuracast.local/radio/8000/admin.h
Administrator Password	RK9tgE9a 📋
Source Password	2LkEpgPf 📋
Relay Password	qRn9RgNz 📋

azuracast application infrastructure

blue boxes indicate docker containers

web container

nginx web server

php primary web application

other tasks handled by this container:
• routine synchronized (cron) tasks
• long-running worker processes
• built-in SFTP service

stations container

supervisord managing process groups for each station

per station

liquidsoap autodj

icecast broadcasting frontend
  or
shoutcast 2 broadcasting frontend

mariadb container

primary web
application database

redis container

application and
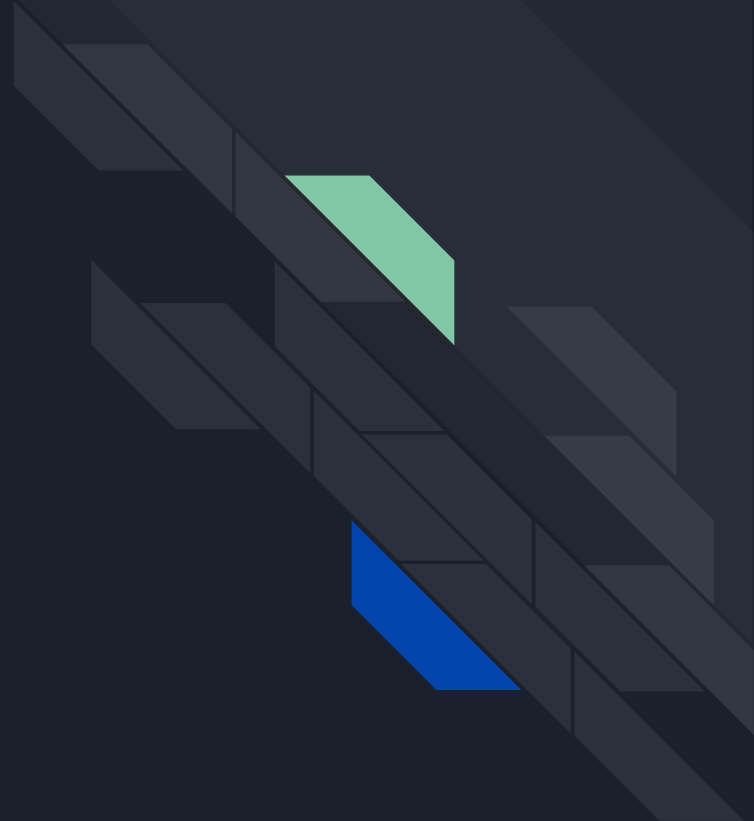session cache

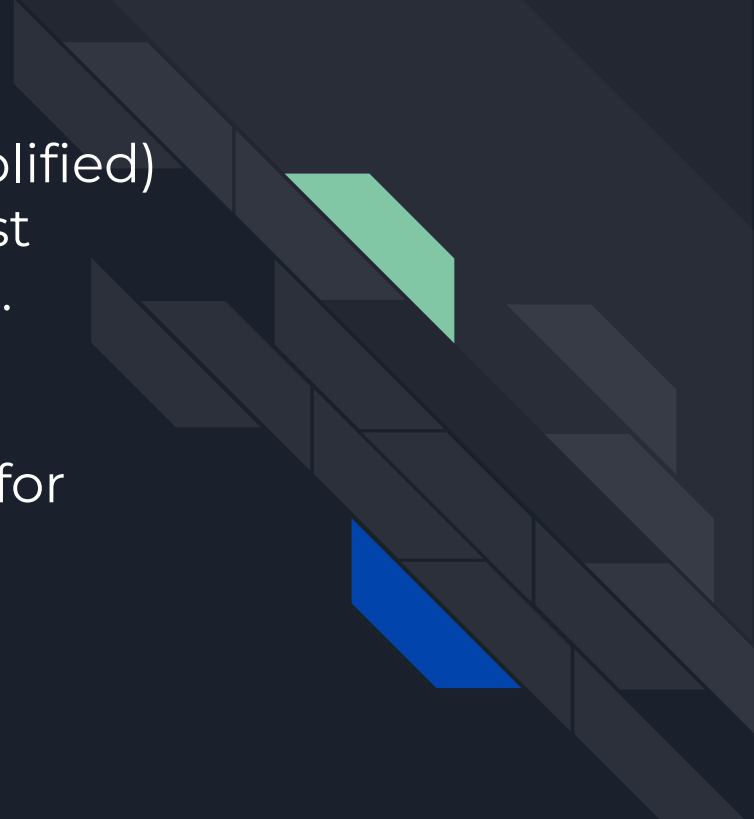AzuraCast and *everything* it installs and maintains is

100%

Free and Open-Source Software

Why Liquidsoap?

- Almost no suitable alternatives (either closed-source or oversimplified)
- Offers full control of the broadcast lifecycle, scheduling, outputs, etc.
- Powered by a script that can be programmatically generated
- Supports incoming connections for DJs/Streamers

How we use Liquidsoap

- OPAM2 builds for Docker AMD64 and Ubuntu 18.04 on Ansible
- Debian packages for Docker ARM64 and Ubuntu 20.04 on Ansible
- Auto-generate configuration from playlists, streamer info, etc.
- Callbacks for metadata, DJ auth, next song, etc.
- Web UI for editing custom configuration spots

```
autodj_is_loading = ref(true)
ignore(autodj_is_loading)

autodj_ping_attempts = ref(0)
ignore(autodj_ping_attempts)

# Track live-enabled status script-wide for fades.
live_enabled = ref(false)
ignore(live_enabled)
```
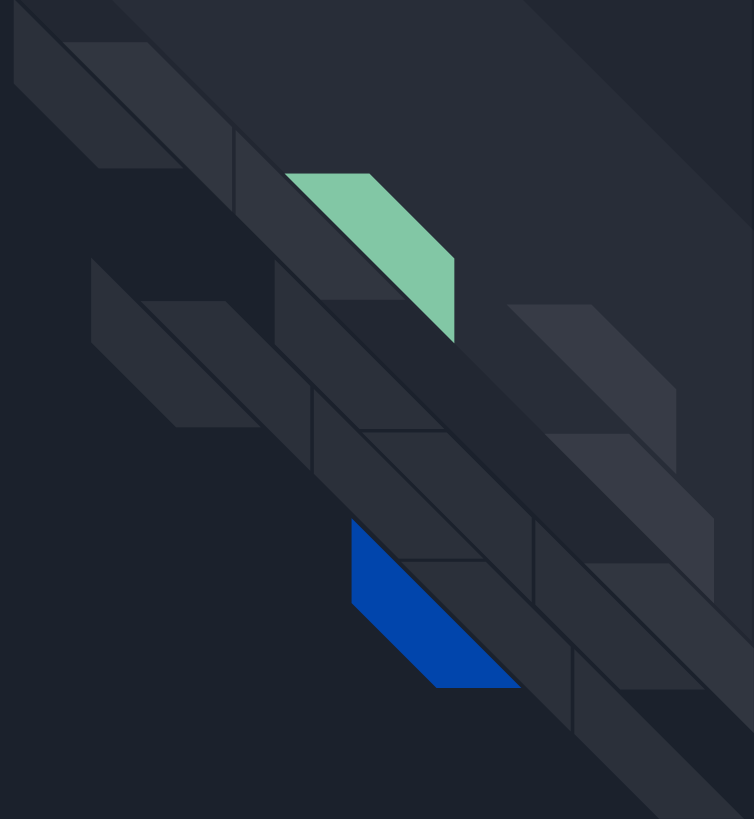
custom_code

```
playlist_default = playlist(id="playlist_default",mime_type="audio/x-mpegurl",mode="randomize",reload_mode="watch","/var/azuracast
playlist_default = cue_cut(id="cue_playlist_default", playlist_default)

playlist_test_schedule_thing = playlist(id="playlist_test_schedule_thing",mime_type="audio/x-mpegurl",mode="randomize",reload_mode
playlist_test_schedule_thing = cue_cut(id="cue_playlist_test_schedule_thing", playlist_test_schedule_thing)

playlist_jingle_thingle = playlist(id="playlist_jingle_thingle",mime_type="audio/x-mpegurl",mode="randomize",reload_mode="watch","
playlist_jingle_thingle = cue_cut(id="cue_playlist_jingle_thingle", playlist_jingle_thingle)
playlist_jingle_thingle = drop_metadata(playlist_jingle_thingle)
```

Lessons Learned

Being Built Upon Liquidsoap

- Issue Overlap
  - Many issues reported to us are Liquidsoap issues
  - User confusion over who fixes what, or where to report issues

- Different Development Styles
  - Frequent changes to the Liquidsoap scripting language

- Project Similarities
  - Similar "Rolling-Release, Tagged-Stable" release cadence
  - Large user base, much smaller maintainer base ("Stadium" arrangement)

Most Impactful Technical Decisions

- Docker as our primary deployment method
    - Significantly decreased support workload
    - Confusion from unfamiliar users
    - A hugely complicated CLI that requires abstraction (Docker Compose) **and** our own utility script to help users
    - Ansible for "Power Users"

- Modern PHP with Standalone Libraries
    - Over 100 Composer package dependencies, 50 more NPM dependencies
    - 5 standalone open-source libraries that operate without AzuraCast
    - HUGE percentage of time spent negotiating "upstream" package updates/changes

Being (and Staying) Open Source

- Life is hard as end-user-facing open-source software
  - Users are unfamiliar with the contributing process
  - Expectations are high, especially in users switching from closed-source/commercial offerings
  - Radio stations are important, so a downtime emergency on the user's part becomes a support emergency on yours

- Staying committed to the principles of open-source
  - Free for everyone, no cost barrier to entry
  - A true "community effort" for the public good
  - Accessibility, inclusivity remain as top priorities

Challenges of Open-Source

- **Sustainability**
    - Project represents thousands of hours per year invested
    - Core maintainer team is small and has not grown in years
    - Support burden is extremely high for team size
    - Donations come through:
        - Patreon
        - GitHub Sponsors
        - Ko-fi, PayPal, Stripe, etc.
    - Total of donations still falls well below FPL for even one person

Attempts at Open-Source Sustainability

- **"Human Services" Professional Services Sector**
  - Little to no interest from users
  - Requires users to submit their own credentials
- **"Software as a Service" Hosted Offering**
  - Has the most potential, but also the greatest risk and overhead
  - Would require pivoting to a completely different business
  - Existing team is developers, not business operators
- **Paid Plugins**
  - Resistance to **any** component of the system being for-profit
  - Determining what ships in the default product and what is an add-on
  - Controlling proliferation of fundamentally open-source code (PHP)

Thank you!

Twitter: **@SlvrEagle23**

AzuraCast

Twitter: **@AzuraCast**

Mastodon: **@AzuraCast@fosstodon.org**

Discord: **discord.gg/azuracast**