# RadioWeave ecosystem

amazon.com

radioweave

Powered by:

amazon web services™

?

?

# API Overview

- JSON REST-based API
- Basic Player - Everything needed to login, get playlists and meta information, and play a radio stream.
- Track Management - Upload/import/remove/reply-to content
- Weave Management - Manage the mix of selected channels
- Data Queries - Search and browse for people, channels, and tracks
- Miscellaneous - Other random stuff

# Request/Response format

**Server requests**

- Most are HTTP GET with standard query parameter syntax including a **jsessionid value (except for login):**

  `http://radioweave.com/v001/api;jsessionid=#####/?param1=value1&param2=value2...`

- Most requests can accept the **heard, skipped, now,** and **loc** query parameters

**Server responses**

- Server will return the "Content-Length" and "Content-Type" HTTP response headers
- For JSON responses, the Content-Type will be "application/json".
- "Accept-Encoding" to "gzip" is supported

# Sample Code (on our hack day link)

- VeryVeryBasicRadioWeavePlayer - Very basic player in a single short HTML page
- Chumby basic player source code - Full Flash Lite source code for the Chumby client
- RadioWeaveMyMusic - Flex source code for this music file uploader
- VeryVeryBasicRadioWeaveUploader - Very basic track uploader in a single short HTML page
- Python examples for uploading tracks - Short python scripts to upload files into RadioWeave: basic, chunked, and with a photo

# Sample Code: RadioWeave My Music

- “RadioWeaveMyMusic” Adobe AIR app (BETA version available now for early adopters)
- Use RWMM to upload songs from your music library on your computer into RadioWeave for your own listening pleasure
- RWMM automatically rotates the songs that are stored on our sever to keep your experience new and interesting

# Sample Code: chumby

- Basic weave player
- Per-track rating
- COMING SOON: “music source” integration to support background play and alarm wakeup

# Things we have to offer fellow hackers

Server & APIs to:

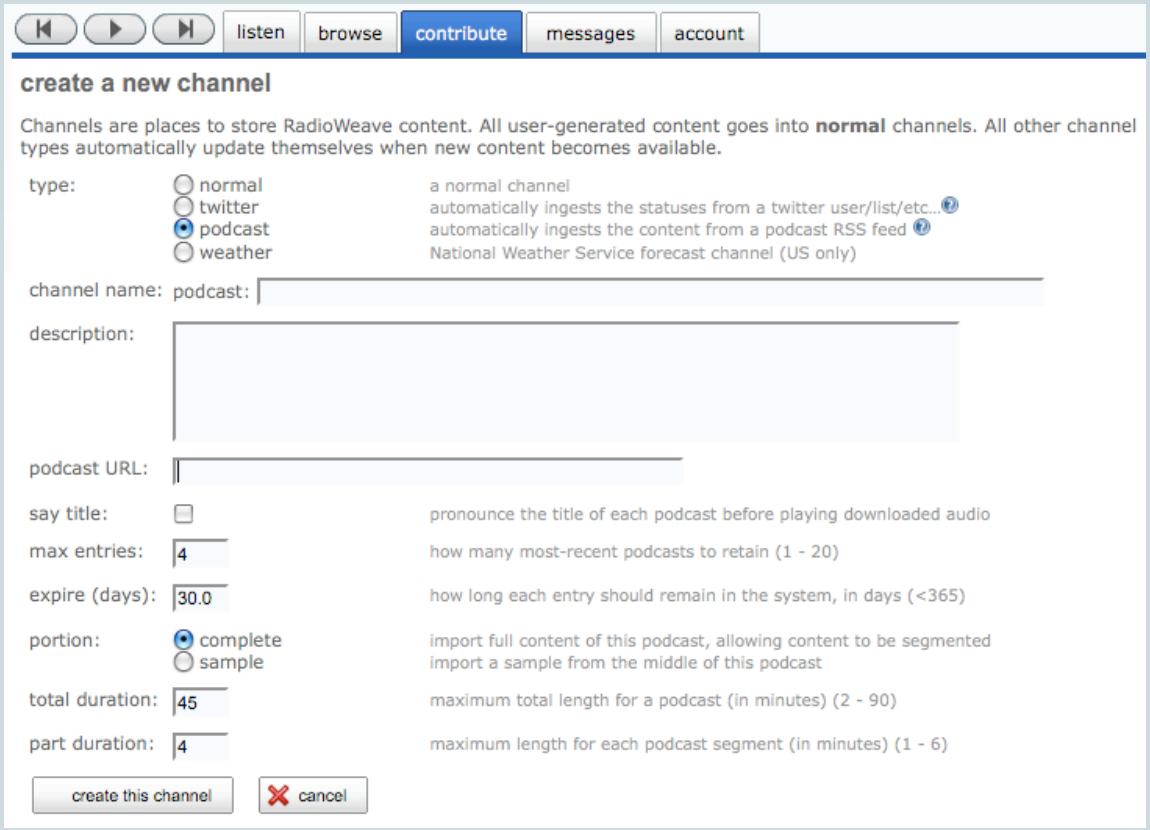
- ingest and combine multiple audio types (source agnostic)
- adjust volumes and compressions so all types of audio stream well together
- chapterize long podcasts, giving room for interstitial music
- import text and convert to audio (using TTS)
- capture listener per-track ratings, skips, repeats

Recording tools:

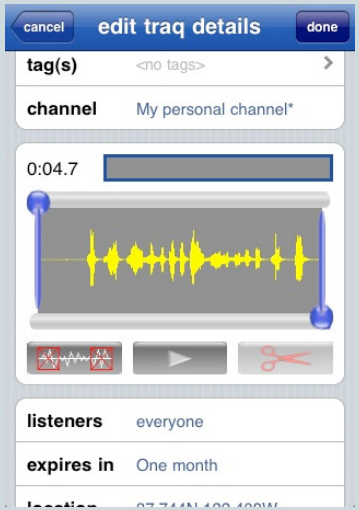
- simple desktop recorder
- very spiffy and unique iPad/iPhone recorders that capture/edit/upload

# Existing channel types

- Creating new micro-channels on radioweave.com:



# Soundbiter (iPhone) App




- 'Cuz you never know when something interesting is going to happen
- Continuously recording, press "bite" to grab the last X minutes of audio
- Simple crop-and-preview audio editor and batch uploader
- Twitter integration for posting uploaded tracks

# Some RW-ish Hack Ideas

- Add any other service as a RadioWeave channel
- Concert Moments using Soundbiter
- Ad-Hoc Event Radio
- Appropriate Interstitial Music
- Threaded Audio Conversations
- Give Content Creators control of their backchannels

# More Hack Ideas

- Audio API Interchange
- Musical mix split-testing
- Audio games
- Scaramouch Scaramouch

# Who are we?

Brent Noorda - brent@radioweave.com

Dale Low - dale@radioweave.com

@radioweave on Twitter

Hacking at the Table Closest To The Free Food