

Station Manager 2

Configurable Automated Netradio
Using Liquidsoap and Icecast

I wanted to run a radio station.








I wanted to focus on the stuff I cared about.

So what did I care about?

(hint: *not code*)

Defining a station...



-  What is the station called?
-  How do listeners connect to it?
-  How do you control and monitor it?
-  What are the different shows?
-  When do shows play across a day, on different days of the week?

Defining its shows...



🏷️ What is each show called?

📁 What music does it play?

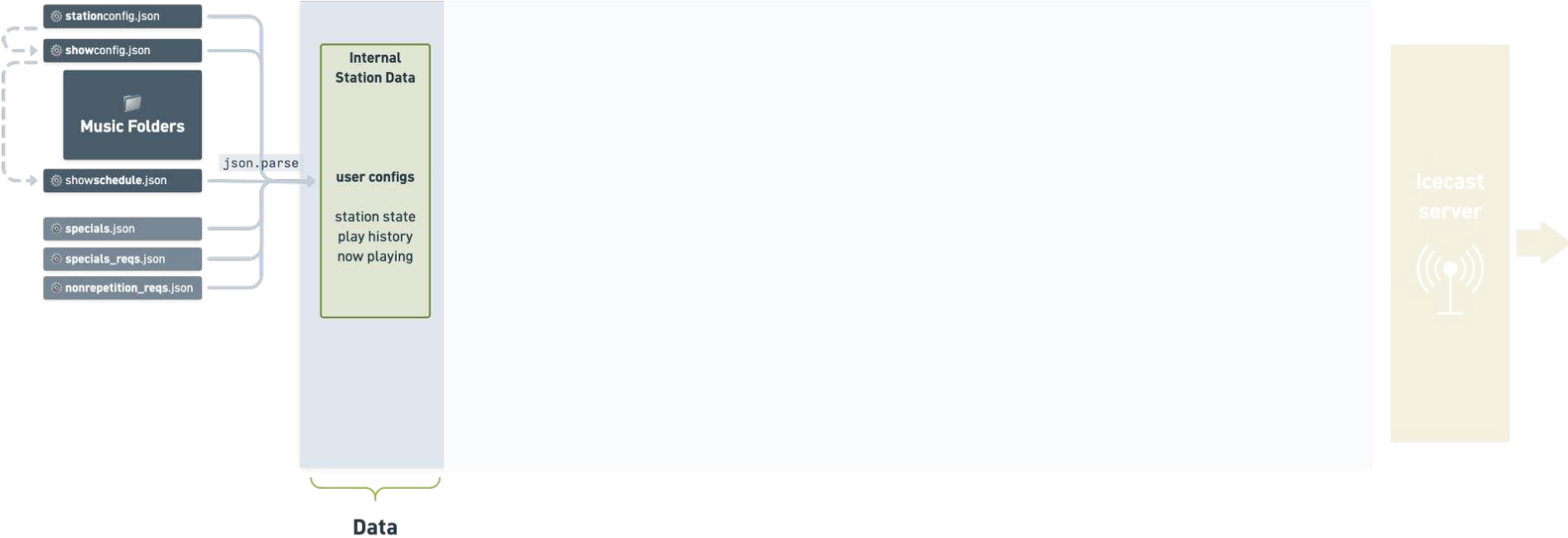
🎤 What does the DJ sound like, and what do they say (and when)?

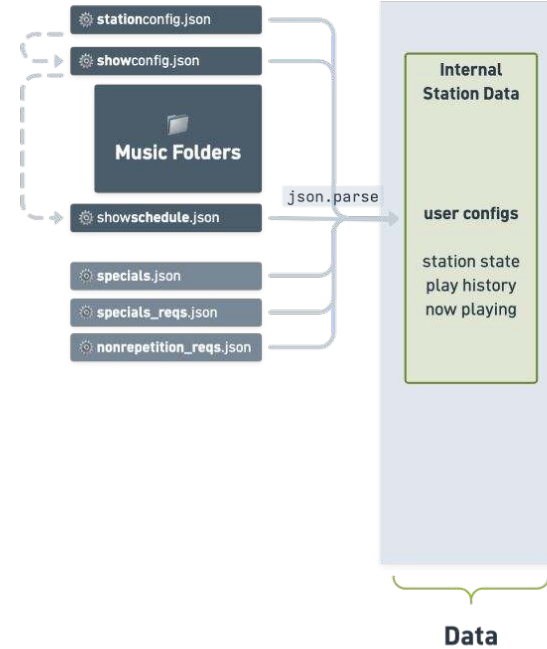
🎵 What kinds of tracks do you want to play within each hour?



Years later...







JSON-based config for...

- **Station** operation and defaults for all shows
- **Show** setup with playlist folders and TTS DJ
- **Schedule** for shows on a daily/weekly basis
- **Play requirements** for *specials* and *non-repetition*



stationconfig.json

Format View As

Key	Class	Value
Root	Dictionary	33 key/value pairs
announcements_path	String	/station/announcements/
cover_default_path	String	/station/default_cover.png
currentshow_path	String	/station/logs/currentShow.log
default_voice	String	Daniel (Enhanced)
http_endpoints	Boolean	YES
http_endpoints_port	Number	1087
icecast_description	String	Testing Station Manager 2
icecast_encoding	String	mp3_192k
icecast_genre	String	Abstract Beats
icecast_host	String	icecast
icecast_mount	String	sm2_test.mp3
icecast_name	String	Station Manager 2 TEST
icecast_password	String	icecast_secret
icecast_port	Number	8000
icecast_url	String	http://localhost:8000/sm2_test
id_announcement	String	You're listening to a Station Manager 2 test stream.
log_path	String	/station/logs/StationManager2_test.log
nonrepetition_reqs_path	String	station_nonrepetition_reqs.json
playlists_root	String	/station/music/
showconfig_path	String	showconfig.json
showschedule_path	String	showschedule.json
specials_path	String	station_specials.json
specials_reqs_path	String	station_specials_reqs.json
telnet_server	Boolean	YES
telnet_server_netmask	String	0.0.0.0
telnet_server_port	Number	1447
telnet_server_timeout	Number	600
trackcheck_timeout	Number	0.5
tracklog_length	Number	12
tracklog_path	String	/station/logs/tracklog.log
tts_announcements_path	String	/tmp/
tts_server_endpoint	String	http://host.doc
voice_type	String	tts_server_ma



Station

stationconfig.json

Key	Class	Value
Root	Dictionary	33 key/value pairs
announcements_path	String	/station/announcements/
cover_default_path	String	/station/default_cover.png
currentshow_path	String	/station/logs/currentShow.log
default_voice	String	Daniel (Enhanced)
http_endpoints	Boolean	YES
http_endpoints_port	Number	1087
icecast_description	String	Testing Station Manager 2
icecast_encoding	String	mp3_192k
icecast_genre	String	Abstract Beats
icecast_host	String	icecast
icecast_mount	String	sm2_test.mp3
icecast_name	String	Station Manager 2 TEST
icecast_password	String	icecast_secret
icecast_port	Number	8000
icecast_url	String	http://localhost:8000/sm2_test
id_announcement	String	You're listening to a Station Manager 2 test stream.
log_path	String	/station/logs/StationManager2_test.log
nonrepetition_reqs_path	String	station_nonrepetition_reqs.json
playlists_root	String	/station/music/
showconfig_path	String	showconfig.json
showschedule_path	String	showschedule.json
specials_path	String	station_specials.json
specials_reqs_path	String	station_specials_reqs.json
telnet_server	Boolean	YES
telnet_server_netmask	String	0.0.0.0
telnet_server_port	Number	1447
telnet_server_timeout	Number	600
trackcheck_timeout	Number	0.5
tracklog_length	Number	12
tracklog_path	String	/station/logs/tracklog.log
tts_announcements_path	String	/tmp/
tts_server_endpoint	String	http://host.doc
voice_type	String	tts_server_ma

showconfig.json

Key	Class	Value
Root	Dictionary	2 key/value pairs
Funkny	Dictionary	8 key/value pairs
DJ_voice	String	Whisper
nonrepetition_reqs_path	String	funky_nonrepetition_reqs.json
playlist_path	String	funky
show_description	String	instrumental hip-hop
show_longname	String	Chillout Beats
shuffle_mode	String	songs
signoff_announcement	String	Thanks for listening to Chillout Beats.
signon_announcement	String	Welcome to Chillout Beats.
Hiphop	Dictionary	8 key/value pairs
DJ_voice	String	Alex
playlist_path	String	hiphop
show_description	String	Abstract hip-hop
show_longname	String	Beats and Rhymes
shuffle_mode	String	songs
signoff_announcement	String	Thanks for listening to Beats and Rhymes
signon_announcement	String	Welcome to Beats and Rhymes
specials_reqs_path	String	hiphop_specials_reqs.json



showschedule.json

Key	Class	Value
Root	Array	2 ordered objects
0	Dictionary	4 key/value pairs
1	Dictionary	4 key/value pairs
days	Array	1 ordered objects
0	String	Sunday
show_hours	Array	2 ordered objects
0	Number	0
1	Number	12
show_names	Array	2 ordered objects
0	String	Funky
1	String	Hiphop
type	String	daysofweek

Shows & Schedule

Station

“Specials” requirements using metadata, e.g....

- Play at least one ★★★★★ track per hour, and try to play it at the top of the hour
- Play at least 3 tracks from the 2010s per hour

station_specials.json

Key	Class	Value
Root	Array	5 ordered objects
0	Dictionary	2 key/value pairs
name	String	Best Of
rules	Array	1 ordered objects
0	Dictionary	3 key/value pairs
filter	String	is number no less than
tag	String	am_rating
value	String	100
1	Dictionary	2 key/value pairs
2	Dictionary	2 key/value pairs
name	String	The 2010s
rules	Array	2 ordered objects
0	Dictionary	3 key/value pairs
filter	String	is number no more than
tag	String	year
value	String	2019
1	Dictionary	3 key/value pairs
filter	String	is number no less than
tag	String	year
value	String	2010
3	Dictionary	2 key/value pairs
4	Dictionary	2 key/value pairs

“Specials” requirements using metadata, e.g....

- Play at least one ★★★★★ track per hour, and try to play it at the top of the hour
- Play at least 3 tracks from the 2010s per hour

station_specials_reqs.json

Key	Class	Value
Root	Array	2 ordered objects
0	Dictionary	3 key/value pairs
per_hour_target	Number	3
special_name	String	The 2010s
try_top_of_hour	Boolean	NO
1	Dictionary	3 key/value pairs
per_hour_target	Number	1
special_name	String	Best Of
try_top_of_hour	Boolean	YES

Key	Class	Value
Root	Array	5 ordered objects
0	Dictionary	2 key/value pairs
name	String	Best Of
rules	Array	1 ordered objects
0	Dictionary	3 key/value pairs
filter	String	is number no less than
tag	String	am_rating
value	String	100
1	Dictionary	2 key/value pairs
2	Dictionary	2 key/value pairs
name	String	The 2010s
rules	Array	2 ordered objects
0	Dictionary	3 key/value pairs
filter	String	is number no more than
tag	String	year
value	String	2019
1	Dictionary	3 key/value pairs
filter	String	is number no less than
tag	String	year
value	String	2010
3	Dictionary	2 key/value pairs
4	Dictionary	2 key/value pairs

“Specials” requirements using metadata, e.g....

- Play at least one ★★★★★ track per hour, and try to play it at the top of the hour
- Play at least 3 tracks from the 2010s per hour

Key	Class	Value
Root	Array	2 ordered objects
0	Dictionary	3 key/value pairs
per_hour_target	Number	3
special_name	String	The 2010s
try_top_of_hour	Boolean	NO
1	Dictionary	3 key/value pairs
per_hour_target	Number	1
special_name	String	Best Of
try_top_of_hour	Boolean	YES

“Non-repetition” requirements using metadata, e.g....

- Never play the *same track* twice in a row
- Never play tracks by the *same artist* within 30 minutes
- Never play tracks from the *same album* within an hour

Key	Class	Value
Root	Array	5 ordered objects
0	Dictionary	2 key/value pairs
name	String	Best Of
rules	Array	1 ordered objects
0	Dictionary	3 key/value pairs
filter	String	is number no less than
tag	String	am_rating
value	String	100
1	Dictionary	2 key/value pairs
2	Dictionary	2 key/value pairs
name	String	The 2010s
rules	Array	2 ordered objects
0	Dictionary	3 key/value pairs
filter	String	is number no more than
tag	String	year
value	String	2019
1	Dictionary	3 key/value pairs
filter	String	is number no less than
tag	String	year
value	String	2010
3	Dictionary	2 key/value pairs
4	Dictionary	2 key/value pairs

“Specials” requirements using metadata, e.g....

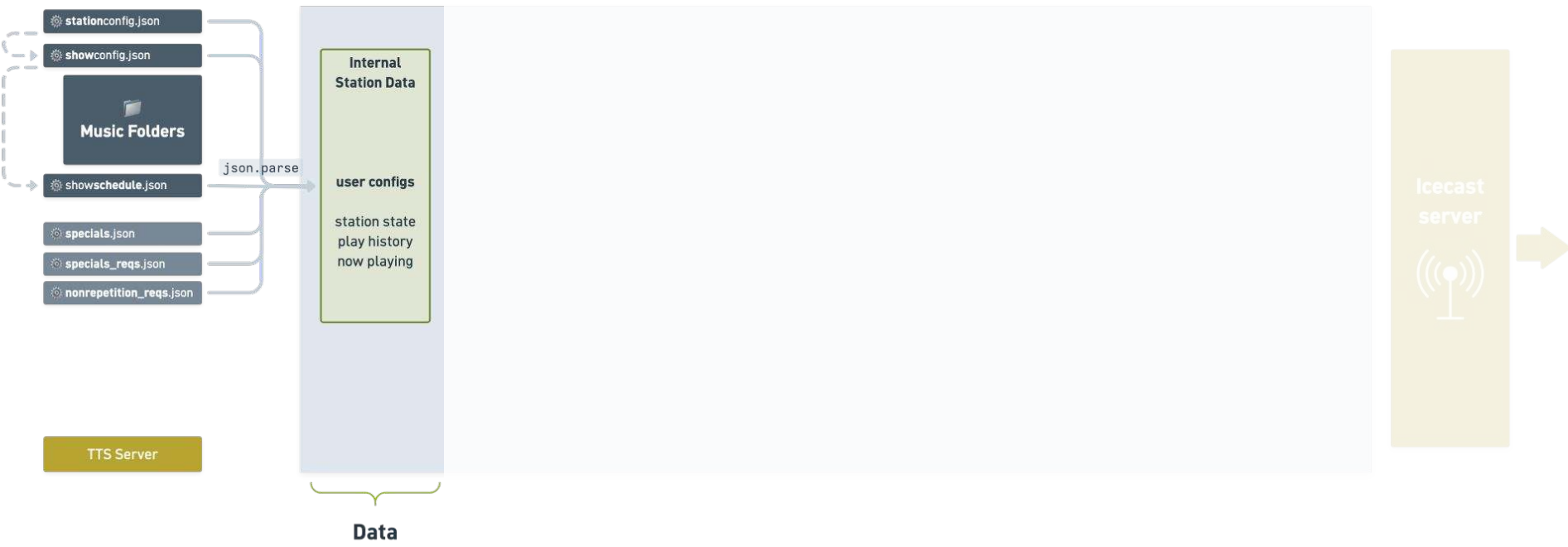
- Play at least one ★★★★★ track per hour, and try to play it at the top of the hour
- Play at least 3 tracks from the 2010s per hour

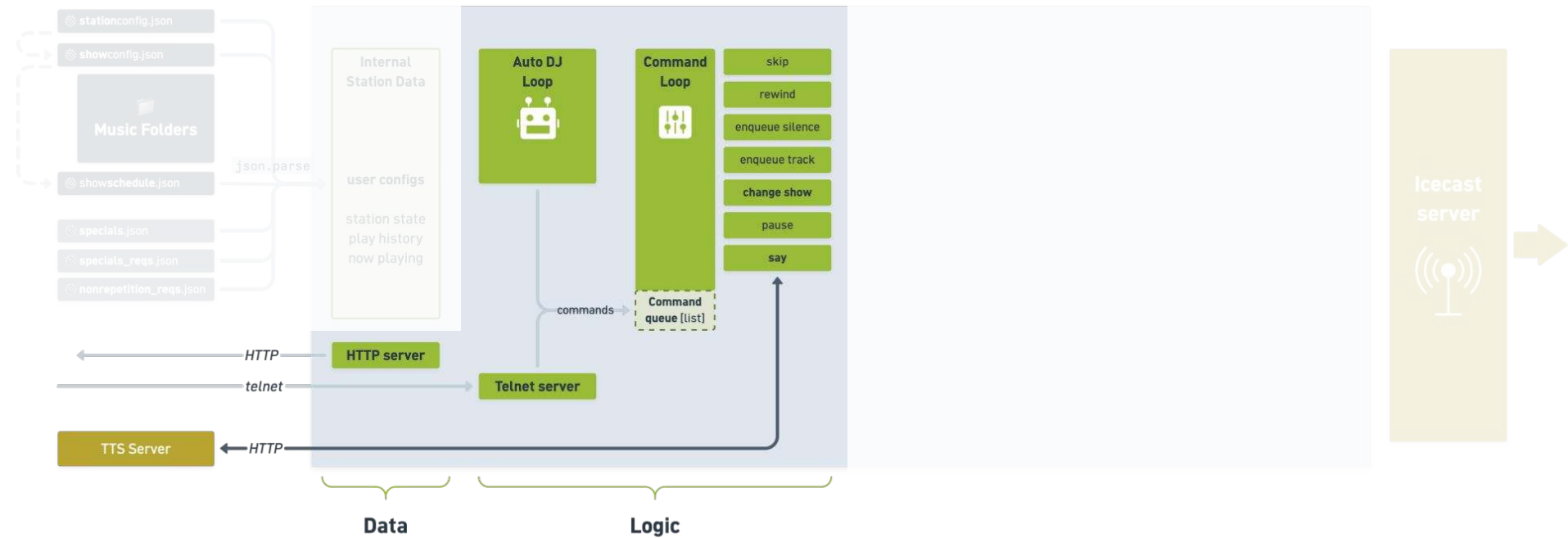
Key	Class	Value
Root	Array	2 ordered objects
0	Dictionary	3 key/value pairs
per_hour_target	Number	3
special_name	String	The 2010s
try_top_of_hour	Boolean	NO
1	Dictionary	3 key/value pairs
per_hour_target	Number	1
special_name	String	Best Of
try_top_of_hour	Boolean	YES

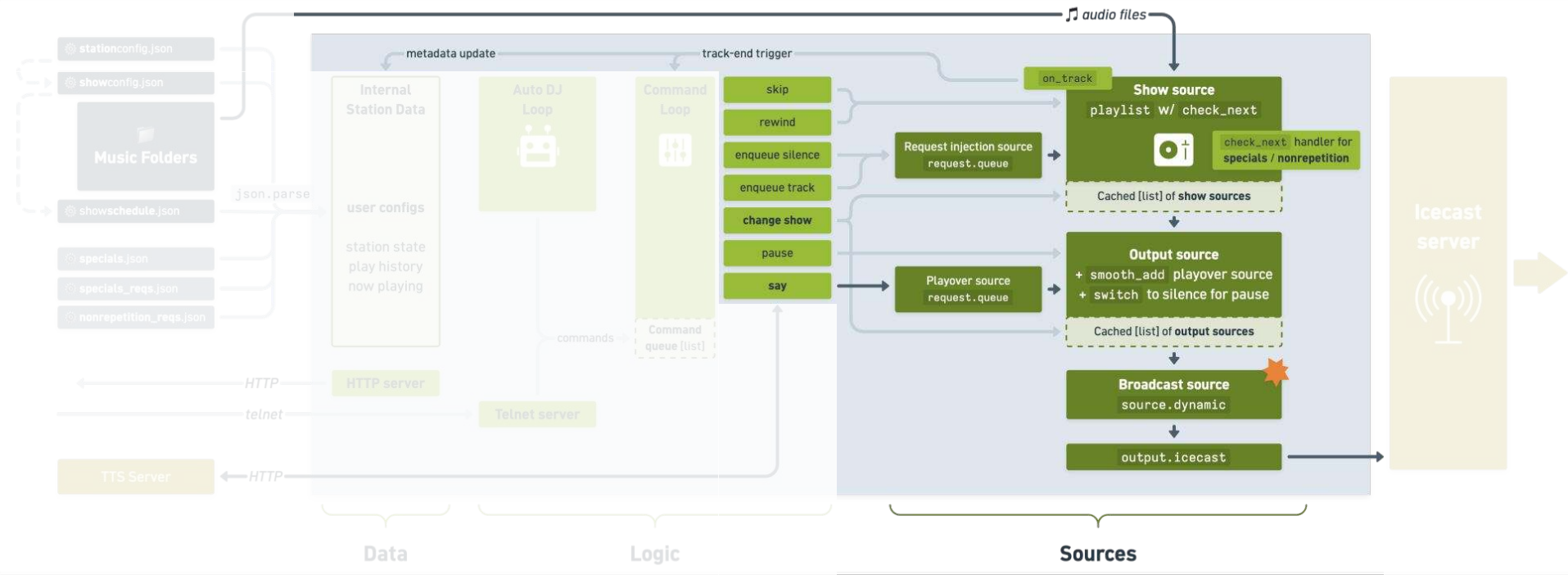
Key	Class	Value
Root	Array	3 ordered objects
0	Dictionary	2 key/value pairs
dont_repeat_within_last	String	track
tag_name	String	filename
1	Dictionary	2 key/value pairs
dont_repeat_within_last	String	30 minutes
tag_name	String	artist
2	Dictionary	2 key/value pairs
dont_repeat_within_last	String	hour
tag_name	String	album

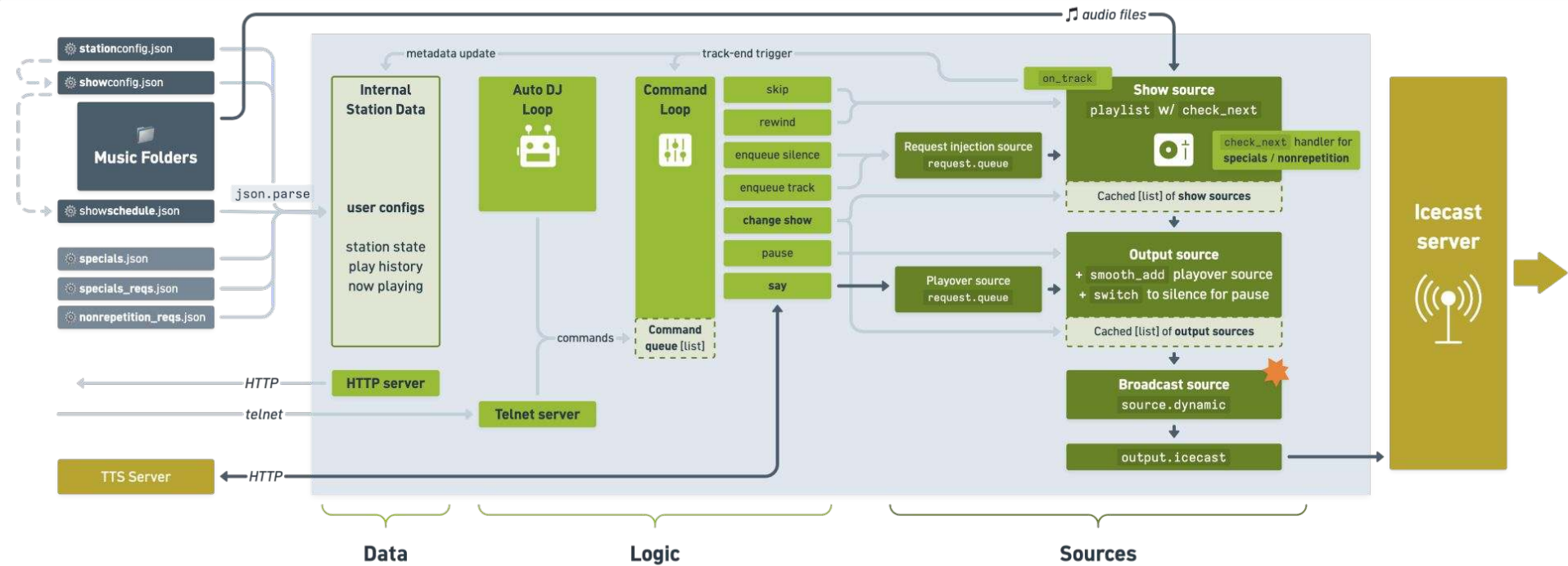
“Non-repetition” requirements using metadata, e.g....

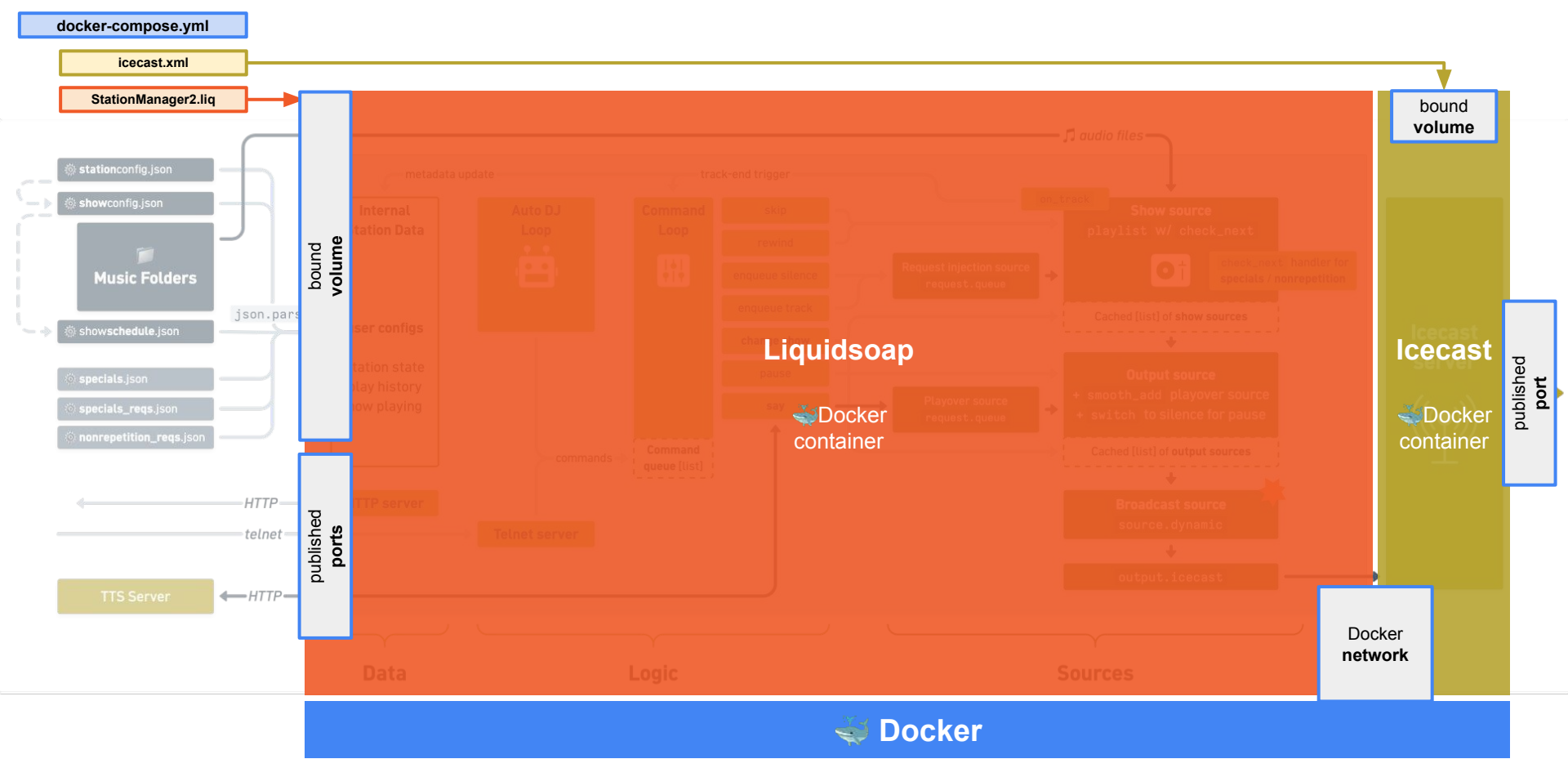
- Never play the same track twice in a row
- Never play tracks by the same *artist* within 30 minutes
- Never play tracks from the same *album* within an hour











Station Manager 2

INTERNET RADIO

<https://stationmanager.tech>

<https://github.com/ymatto/Station-Manager-2>

BASSCADET

<https://basscadet.net>

