

REPEATER-DB → CODEPLUGS

Lukas OE3LRT

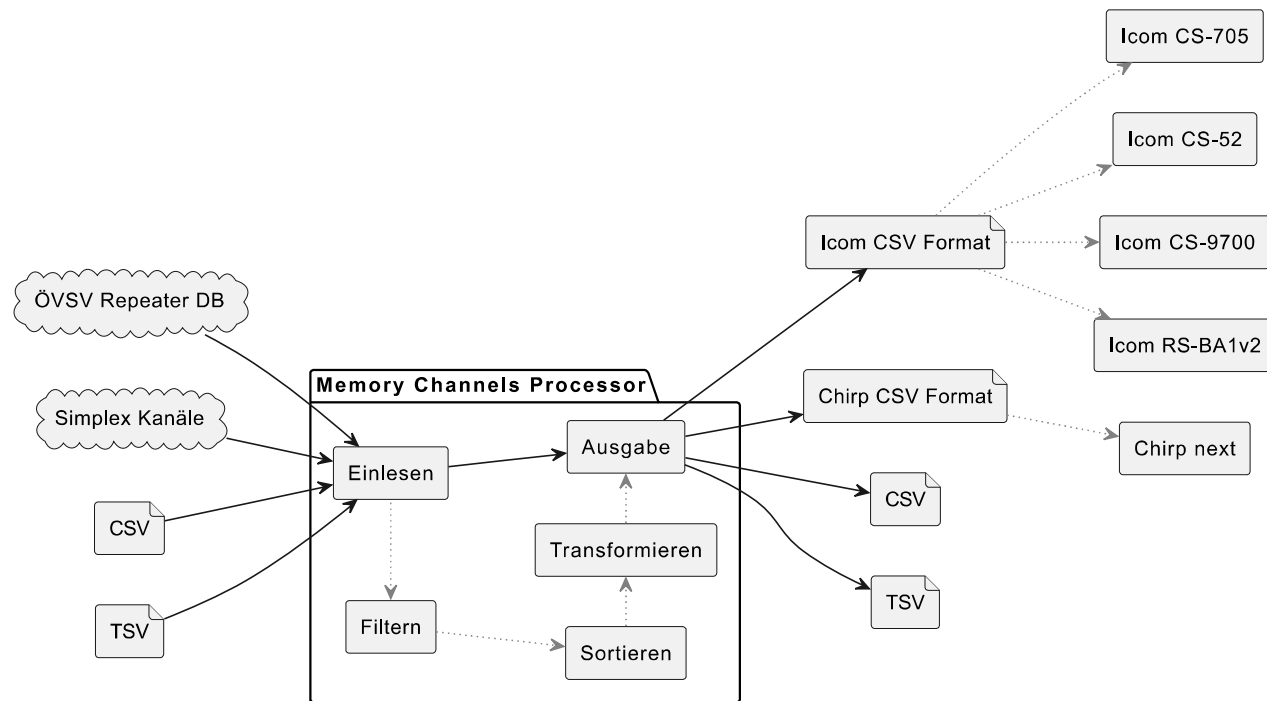
IDEE

Erstellung der Funkgeräte-Konfiguration (Repeater & Kanäle) ...

- ... möglichst aktuell
- ... soweit wie möglich automatisiert
- ... möglichst unkompliziert in der Bedienung
- ... reproduzierbar
- ... konsistent für mehrere Funkgeräte
- ... mit Möglichkeit zur Individualisierung
(z.B. Einträge zu oder "Hausfrequenzen")

Umsetzung im `Memory Channels Processor`

KONZEPT



TOOLSET: ÖVSV-REPEATER-DB? CODEPLUG?

ÖVSV-Repeater-DB

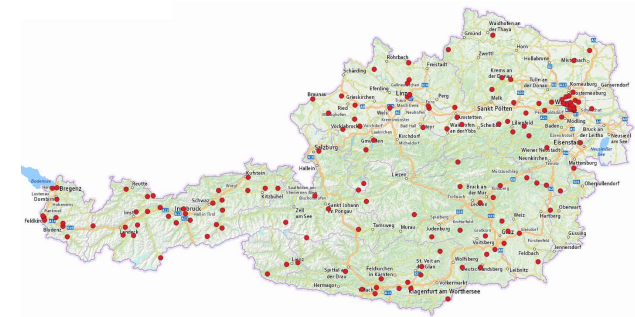
- Datenbank von österreichischen Repeatern mit
 - Frequenzen
 - Betriebsarten
 - Subtönen
 - Koordinaten
 - ...
- eine **API** bietet Daten zur Weiterverarbeitung an

Codeplug

Eine am Computer erstellte Konfiguration, die mit spezieller Software ans Funkgerät übertragen wird



repeater.oevsv.at/de



TOOLSET: PYTHON?

- interpretierte Programmiersprache
- plattformunabhängig
- populär
- gut lesbar



python.org

TOOLSET: GIT? GITLAB? CI?

Git

Software zur Versionsverwaltung von Dateien

- Versionierung des Codes
- Verteilte Entwicklung

GitLab

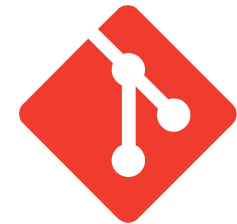
Webanwendung zur Versionsverwaltung für Softwareprojekte auf Git-Basis

- Hosting des Codes
- Bereitstellen der Website & Codeplugs (über *GitLab Pages*)

CI

Continuous Integration
= Prozess des fortlaufenden Zusammenfügens von Komponenten zu einer Anwendung

- Automatisiertes Ausführen des **Memory Channels Processor**
- Bauen und Bereitstellen der **Codeplugs**



git-scm.com



gitlab.com

HIGHLIGHTS (1)

- Anbindung an die ÖVSV Repeater-DB per API
- Mehrere Datenquellen parallel verarbeiten
- Unterstützung für
 - FM Repeater
 - D-Star Repeater
- Filtern nach ...
 - ... Bändern
 - ... Betriebsarten

HIGHLIGHTS (2)

- Sortierung nach ...
 - ... Name
 - ... Rufzeichen
 - ... Frequenz
 - ...
- vordefinierte Konfigurationen werden jede Nacht automatisch generiert
 - alle Änderungen aus der Repeater-DB
 - tagesaktuell
 - gleich zum Download verfügbar

WEBSITE

Memory Channels Processor

Search the docs Home Links GitLab Repo

Memory Channels Processor Memory Channels Processor / Home Edit this Page

Home

- Codeplugs
 - Icom
 - Chirp
- Import & Export
 - CSV
 - TSV
- Softwareentwicklung

Memory Channels Processor

NOTE

Bitte beachten Sie, dass es sich bei diesem Dokument um einen Entwurf handelt, der noch in Arbeit ist. Es wird der Öffentlichkeit nur zu Informationszwecken zur Verfügung gestellt. Da es sich um ein unfertiges Dokument handelt, fehlen einige Teile oder werden gerade überarbeitet.

Das Programm `Memory Channels Processor` bietet die Möglichkeit Simplex- und Relais-Kanäle aus verschiedenen Quellen für verschiedene Geräte (z.B. Icom IC-705, ID-52E oder IC-9700) oder für Software (z.B. Chirp) zur Weiterverarbeitung aufzubereiten.

Die ursprüngliche Umsetzung^[1] wurde mit dem Ziel entwickelt die österreichischen 2m/70cm FM und D-Star Amateurfunk Relaiskanäle^[2] möglichst automatisiert für den Import für das Icom IC-705 und ID-52E bereitzustellen.

Die zugrundeliegende Idee wurde in dieser Applikation weiterentwickelt und auf andere technische Beine gestellt.

Mittlerweile gibt es auch die Möglichkeit Daten von anderen Quellen zu importieren und Daten dieser Quellen zu mischen und auch eigens definierte Datensätze hinzuzufügen.

Memory Channels Processor latest

Contents

- Downloads
- Bedienung
- Softwareentwicklung

oe3lrt.gitlab.io/memory-channels-processor

CODEPLUGS

Memory Channels Processor

Search the docs Home Links GitLab Repo

Memory Channels Processor / Codeplugs / Icom

Memory Channels Processor / Codeplugs / Icom

NOTE

Das Dateiformat der Codeplugs für Icom IC-705 und ID-52E kann auch im Memory Channel Editor der PC-Software RS-BA1v2 Remote Control importiert werden.

Codeplugs

Die folgenden Codeplugs sind für die gängigsten Einsatzszenarien gedacht, wurden bereits automatisch erstellt und sind zum Download verfügbar.

Codeplugs für Icom IC-705 & ID-52E

Name	Datei	Band					Modus			Quelle
		23cm	70cm	2m	6m	10m	FM	D-Star	ÖVSV	Simplex-Kanäle
OE 2m Repeaters	icom_ic705_id52_fm_2m_gen_oevsv-repeater-db 1)			X			X		X	
OE 2m Repeaters + Channels	icom_ic705_id52_fm_2m_gen_oevsv-repeater-db_fm-channels-iaru-r1 1)			X			X		X	X
OE 2m+70cm Repeaters	icom_ic705_id52_fm_2m_70cm_gen_oevsv-repeater-db 1)		X	X			X		X	
OE 2m Channels	icom_ic705_id52_fm_2m_gen_fm-channels-iaru-r1 1)			X			X			X
OE 2m+70cm Channels	icom_ic705_id52_fm_2m_70cm_gen_fm-channels-iaru-r1 1)		X	X			X			X
OE 70cm Repeaters	icom_ic705_id52_fm_70cm_gen_oevsv-repeater-db 1)		X				X		X	
OE 70cm Repeaters + Channels	icom_ic705_id52_fm_70cm_gen_oevsv-repeater-db_fm-channels-iaru-r1 1)		X				X		X	X
OE 2m+70cm Repeaters + Channels	icom_ic705_id52_fm_2m_70cm_gen_oevsv-repeater-db_fm-channels-iaru-r1 1)		X	X			X		X	X

Memory Channels Processor latest

Contents

- Icom IC-705 und ID-52E (sowie RS-BA1V2)
- Codeplugs
- Beispiele
- Icom IC-9700
- Codeplugs
- Beispiele
- Fehler beim Importieren von D-Star

oe3lrt.gitlab.io/memory-channels-processor/codeplug_icom.html

KOMMANDOZEILE (1)

```
$ memory-channels-processor --source "oevsv-repeater-db"  
  --band 2m  
  --type "fm"  
  --output-file "fm_2m_repeater.csv"  
  --output-format="icom-ic-705"
```

KOMMANDOZEILE (2)

```
$ memory-channels-processor --source "oevsv-repeater-db"  
  --source "csv"  
  --band 2m  
  --type "fm"  
  --csv-input-file "fm_additional.csv"  
  --output-file "fm_2m_gen_repeater_additional.csv"  
  --output-format="icom-ic-705"
```

WHAT'S NEXT?

- Support für DMR-Repeater
- mehr Filtermöglichkeiten (z.B. Wildcards od. RegEx)
- Unit-Tests
- Fehlerhandling verbessern
- Unterstützung für weitere Codeplugs
(siehe oevsv.at/funkbetrieb/amateurfunkfrequenzen/ukw-referat/maps)

HELP WANTED!

LINKS

ÖVSV-Repeater-Datenbank:

- <https://repeater.oevsv.at/de/>
- <https://www.oevsv.at/funkbetrieb/amateurfunkfrequenzen/ukw-referat/maps/>

Memory Channels Processor:

- <https://oe3lrt.gitlab.io/memory-channels-processor>
- <https://gitlab.com/oe3lrt/memory-channels-processor>
- <https://gitlab.com/oe3lrt/memory-channels-processor-presentation>
- <https://oe3lrt.gitlab.io/memory-channels-processor-presentation>