



LogRunner

Benutzerhandbuch

Version **0.9.3b61** (Beta)

Stand 13.09.2026

Sprache Deutsch

Inhalt

1. Übersicht & Verwendungszweck

Hauptfunktionen im Überblick

2. Systemvoraussetzungen & Installation

2.1 Voraussetzungen

2.2 Abhängigkeiten (aus pyproject.toml)

2.3 Installation der Windows-Beta

2.4 LogRunner auf dem Raspberry Pi

2.5 Start aus dem Quellcode

2.6 Der Datenordner

2.7 Das Startfenster und die Lizenz

3. Benutzeroberfläche & Navigation

3.1 Das Hauptfenster

3.2 Die Ansicht

4. Funktionsbeschreibung & Workflows

4.1 Workflow 1 – Ein QSO loggen

4.2 Workflow 2 – Datenaustausch, Import, Export, Schnittstellen

4.3 Workflow 3 – Contestbetrieb

4.4 Workflow 4 – Nachsehen, was schon gearbeitet ist

4.5 Workflow 5 – Die Karte

4.6 Workflow 6 – Bandbedingungen

4.7 Workflow 7 – Vorgaben für QSOs

4.8 Workflow 8 – Anzeige anpassen

4.9 Workflow 9 – Alle QSOs löschen

5. Einstellungen & Konfiguration

5.1 Wo die Einstellungen liegen

5.2 Die Abschnitte von config.toml

5.3 Wichtige Einzeloptionen

5.4 Umgebungsvariable

5.5 Kommandozeile

5.6 Tastenkombinationen

6. Fehlerbehebung (Troubleshooting)

6.1 Wo die Protokolle liegen

6.2 Startprobleme

6.3 Logbuch und Daten

6.4 Wavelog-Abgleich

6.5 Rufzeichen-Abfrage

6.6 DX-Cluster

6.7 TRX-Steuerung und Wasserfall

6.8 WSJT-X-Übernahme

6.9 Bildaufnahme

6.10 LiveLink

6.11 Wenn gar nichts hilft

Anhang: Hilfe und Unterstützung

Anhang: Daten und Lizenzen

1. Übersicht & Verwendungszweck

LogRunner ist ein portabler Offline-Logger für den Amateurfunk. Er führt das Stationslogbuch, hält es in einer einzelnen SQLite-Datei und arbeitet dabei vollständig ohne Internetverbindung. Alle Online-Dienste – Wavelog-Abgleich, DX-Cluster, Rufzeichen-Abfrage, Bandbedingungen – sind Zugaben, sind ab Werk **ausgeschaltet** und lassen sich einzeln zuschalten.

Gedacht ist das Programm für Funkamateure, die ihr Logbuch selbst in der Hand behalten wollen: für den Alltagsbetrieb im Shack, für Portabelbetrieb ohne Netz, für Contest-Wochenenden und für Stationen, die mehrere Bildschirme oder mehrere PCs nebeneinander betreiben.

Der gesamte Datenbestand liegt in **einem** Ordner (~/LogRunner/). Dieser Ordner lässt sich auf einen USB-Stick kopieren und auf einem anderen Rechner weiterbetreiben – wer ihn sichert, hat alles gesichert.

Hauptfunktionen im Überblick

- **QSO-Erfassung** über eine Eingabemaske mit sieben Reitern (QSO, Station, Allgemeines, Contest, Sat, Notizen, QSL) – dieselbe Feldaufteilung wie in Wavelog, dahinter durchgehend ADIF-Felder.
- **Live-Auflösung des Rufzeichens** über `cty.dat`: Land, Kontinent, CQ- und ITU-Zone sowie die DXCC-Nummer werden beim Loggen automatisch eingetragen.
- **Log-Tabelle** mit Suche und Filter, frei wählbaren Spalten (81 stehen zur Auswahl, nach den Reitern der Eingabemaske gruppiert), Sortierung nach jeder Spalte und Bearbeiten/Löschen je QSO.
- **Gearbeitete Bänder**: Bandmatrix je Rufzeichen mit den Zuständen *gearbeitet* / *bestätigt* / *noch nicht*, wahlweise nach Betriebsart aufgeschlüsselt.
- **Statistik**: Kacheln, Zeitverlauf, Band-gegen-Mode-Matrix, Auswertung nach Kontinent, Jahr, Standort und Operator sowie IOTA-Fortschritt.
- **Weltkarte offline** in drei Ansichten (flache Karte, Globus, Azimutkarte) mit Graulinie, DXCC-Präfixen, CQ-/ITU-Zonen, Locator-Raster, Ortsnamen, IOTA-Inselgruppen, DX-Spots und den Log-Einträgen.
- **DX-Cluster** über Telnet mit Tabelle, vertikaler Band-Map und Meldungs- statistik, Filtern mit Jokerzeichen, gespeicherten Filtersätzen und akustischen Alarmregeln.
- **Contestbetrieb** mit Sitzung, laufender Nummer, Dupe-Warnung, einer tastaturgetriebenen Contest-Zeile und Cabrillo-Einreichung.
- **TRX-Steuerung** des Transceivers über Icom CI-V, Hamlib (`rigctl`) oder Omnirig – einschließlich Wasserfall aus dem Bandscope von Icom-Geräten.
- **WSJT-X-Übernahme**: was in WSJT-X geloggt wird, steht Sekunden später im Log.
- **Gerätebild über eine HDMI-Capture-Karte**, wahlweise im Netz angeboten.
- **ADIF-Import und -Export** mit tolerantem Parser und Dublettenlogik.
- **Wavelog-/Cloudlog-Abgleich** über API v1 und v2 mit Offline-Warteschlange.
- **LiveLink** mehrerer PCs im LAN (ein Hub, beliebig viele Clients), bei der jeder Rechner nur einen Bereich formatfüllend zeigen kann.
- **POTA-, SOTA- und IOTA-Listen** offline durchsuchbar, dazu die Super-Check-Partial-Liste für Rufzeichenvorschläge.
- **Sechs Sprachen** (Deutsch, English, Español, Français, Italiano, Português), umschaltbar ohne Neustart, sowie helles und dunkles Thema.

2. Systemvoraussetzungen & Installation

2.1 Voraussetzungen

Voraussetzung	Anforderung
Betriebssystem	Windows 10/11 (fertige Beta). macOS und Linux laufen aus dem Quellcode.
Anzeige-Engine	Microsoft Edge WebView2 Runtime , mindestens Chromium-Version 111
Python (nur aus dem Quellcode)	3.11 oder neuer
Node.js (nur zum Bauen der Oberfläche)	aktuelle LTS-Fassung
Plattenplatz	rund 50 MB für das Programm, dazu der Datenordner

⚠ **Achtung:** Ohne die WebView2-Runtime startet LogRunner nicht. Das Programm erkennt das und sagt es in einem Meldungsfenster; es fällt bewusst **nicht** stillschweigend auf die alte Internet-Explorer-Anzeige zurück, in der die Oberfläche schwarz bliebe. Abhilfe: den *Evergreen Standalone Installer (x64)* für WebView2 von Microsoft installieren und LogRunner neu starten.

2.2 Abhängigkeiten (aus `pyproject.toml`)

Paket	Mindestversion	Wofür
<code>pywebview</code>	5.1	Das Fenster und die Brücke zwischen Oberfläche und Python
<code>tomli-w</code>	1.0	Schreiben von <code>config.toml</code>
<code>pyserial</code>	3.5	Serielle Leitung zum Icom-CI-V-Bus (nur bei Bedarf geladen)
<code>websockets</code>	12.0	LiveLink zwischen mehreren PCs
<code>zeroconf</code>	0.131.0	Automatisches Finden des Hubs per mDNS

Optional:

Paket	Wofür
<code>comtypes ≥ 1.2 (pip install logrunner[omnirig])</code>	Nur für die Omnirig-Anbindung, nur unter Windows
<code>pytest ≥ 8.0, ruff ≥ 0.16, mypy ≥ 1.11, pyyaml ≥ 6.0, pyinstaller ≥ 6.11, pillow ≥ 10.0 ([dev])</code>	Tests, Prüfwerkzeuge und Release-Bau

Externe Programme, die LogRunner **nicht** mitbringt:

- **ffmpeg** – nur für die Bildaufnahme über eine Capture-Karte. Ein Knopf im Panel *Bildaufnahme* lädt es unter Windows selbst herunter (rund 110 MB, einmalig).
- **rigctld** (Hamlib) beziehungsweise **Omnirig** – nur für die entsprechende CAT-Anbindung.

2.3 Installation der Windows-Beta

1. Das Archiv `LogRunner-<Version>-win64.zip` **vollständig** entpacken – der Ordner enthält `LogRunner.exe` und daneben das Verzeichnis `_internal`.
2. Den entpackten Ordner an eine beliebige Stelle legen (Festplatte, USB-Stick, Netzlaufwerk). Eine Installation im eigentlichen Sinn findet nicht statt, es werden keine Registry-Einträge geschrieben.
3. `LogRunner.exe` starten – per Doppelklick oder über eine Verknüpfung.

⚠ Achtung: Das Programm besteht aus einem Ordner, nicht aus einer einzelnen Datei. Wird nur LogRunner.exe herauskopiert, fehlt der Web-Teil, und der Start bricht mit der Meldung „Der Web-Teil des Programms fehlt“ ab.

2.4 LogRunner auf dem Raspberry Pi

Getestet auf einem **Raspberry Pi 5 (8 GB) mit Raspberry Pi OS / Debian 13 (trixie), 64 bit**. Das Archiv heißt LogRunner-<Version>-linux-arm64.zip; es läuft nur auf 64-bit-ARM, nicht auf einem PC.

1. Entpacken – am besten dort, wo das Archiv liegt, zum Beispiel in ~/Downloads. Wie unter Windows: das ganze Archiv, nicht nur die Programmdatei.

```
cd ~/Downloads
unzip LogRunner-0.9.3b61-linux-arm64.zip
```

2. Installieren – mit dem Skript aus dem Archiv:

```
bash LogRunner/install.sh
```

Das Skript erledigt in einem Zug, was sonst von Hand ginge (siehe unten), und fragt nur, wo es etwas entscheiden muss:

Schritt	Was geschieht
Prüfen	Richtiger Rechner (64-bit-ARM), nicht als root gestartet, kein Overlay-Dateisystem – ist es an, bricht das Skript ab und sagt, wie es abzuschalten ist, denn sonst läge alles nur im Arbeitsspeicher und wäre nach dem nächsten Neustart weg, samt Logbuch
Systempakete	WebKitGTK (zeigt die Oberfläche) und gio (startet die Begleitfenster) werden nachinstalliert, wenn sie fehlen. Hamlib und ffmpeg nur, wenn man danach gefragt Ja sagt. Dafür wird das sudo -Passwort gebraucht
Transceiver	Der eigene Benutzer kommt in die Gruppe dialout, damit LogRunner den seriellen Anschluss öffnen darf. Einmal ab- und wieder anmelden, das Skript erinnert daran
Programm	Kopiert nach ~/Programme/LogRunner. Ist dort schon eine Fassung, wird sie beiseitegeschoben , nicht gelöscht; die letzten zwei beiseitegeschobenen bleiben stehen, ältere räumt das Skript auf
Startmenü	Ein Eintrag unter Hamradio ▶ LogRunner – zum Starten oder zum Anheften ans Panel

Ein Update geht genauso: alle LogRunner-Fenster schließen (der Hub nimmt seine Begleitfenster mit), das neue Archiv entpacken, bash LogRunner/install.sh. Läuft LogRunner noch, bricht das Skript ab, statt das Programm unter den offenen Fenstern auszutauschen. Der **Datenordner ~/LogRunner** – Logbuch, Einstellungen, Protokolle – wird dabei nie angefasst.

Weitere Möglichkeiten zeigt bash LogRunner/install.sh --help: ein anderer Ort (--prefix), ohne Nachfragen (-yes), ohne Systempakete (--no-apt) oder ohne Aufräumen (--keep-all).

Von Hand

Wer lieber selbst Hand anlegt, braucht dieselben Schritte einzeln.

Anzeige-Bibliothek und gio – LogRunner bringt GTK mit, WebKitGTK kommt vom System:

```
sudo apt install gir1.2-webkit2-4.1 libglib2.0-bin
```

Wer den Transceiver über Hamlib statt über CI-V anspricht, nimmt `libhamlib-utils` dazu; für die Bildaufnahme `ffmpeg`. Fehlt WebKitGTK, sagt LogRunner beim Start selbst, welches Paket zu installieren ist.

Programm an seinen Platz – den entpackten Ordner verschieben:

```
mkdir -p ~/Programme
mv ~/Downloads/LogRunner ~/Programme/
chmod +x ~/Programme/LogRunner/LogRunner
```

⚠ **Nicht nach `~/LogRunner` entpacken oder verschieben.** Genau dort liegt der **Datenordner** – Logbuch, `config.toml`, Protokolle. Programm und Daten landeten sonst im selben Verzeichnis; ein Update, das den Programmordner ersetzt, nähme das Logbuch mit. `~/Programme` ist nur ein Vorschlag, jede andere Stelle geht auch – nur eben nicht diese eine.

Fensterstellen unter Wayland. Der Raspberry-Pi-Desktop (labwc) ist ein Wayland-Compositor, und dort kann sich kein Fenster seine Position merken – auch LogRunner nicht; Größe und Ansicht schon. Zwei Wege gibt es trotzdem: eine Fensterregel des Compositors, oder LogRunner mit `GDK_BACKEND=x11` starten. Beides steht bei *Startzustand merken*.

3. Starten – per Doppelklick im Dateimanager oder auf der Konsole:

```
~/Programme/LogRunner/LogRunner
```

Ein Symbol für den Startknopf legt diese Datei an – einmal schreiben, dann steht LogRunner im Menü unter *Hamradio* und lässt sich von dort starten oder ans Panel heften:

```
cat > ~/.local/share/applications/logrunner.desktop <<'ENDE'
[Desktop Entry]
Type=Application
Name=LogRunner
Comment=Offline-Logbuch für den Amateurfunk
Exec=/home/USER/Programme/LogRunner/LogRunner
Path=/home/USER/Programme/LogRunner
Icon=/home/USER/Programme/LogRunner/_internal/logrunner/assets/logo.png
Terminal=false
Categories=HamRadio;Network;
ENDE
```

`USER` durch den eigenen Benutzernamen ersetzen – die Zeilen müssen absolute Pfade tragen, `~` versteht eine `.desktop`-Datei nicht.

Der Transceiver meldet sich als `/dev/ttyACM0`, nicht als `/dev/ttyUSB0` – ein IC-7300MK2 ist ein USB-Verbundgerät und kommt über den CDC-ACM-Treiber. Wer dort nach `ttyUSB` sucht, sucht falsch. Der eigene Benutzer muss in der Gruppe `dialout` sein:

```
sudo usermod -aG dialout $USER
```

Danach einmal ab- und wieder anmelden.

Was auf einem Pi anders ist als unter Windows:

Anzeige	WebKitGTK statt WebView2
---------	--------------------------

Verknüpfungen für Begleitfenster	<code>.desktop</code> -Dateien statt <code>.lnk</code>
Transceiver-Anbindung	CI-V und Hamlib; Omnirig gibt es dort nicht
Serielle Anschlüsse	<code>/dev/ttyACM*</code> und <code>/dev/ttyUSB*</code>

Gemessen an einem Pi 5 mit angeschlossenem IC-7300MK2: eine Statusabfrage des Geräts braucht 3,7 ms, die volle Reglerleiste 33 ms, der Wasserfall läuft mit 3,3 Durchläufen je Sekunde ohne einen einzigen Aussetzer – das Tempo setzt dabei der Transceiver, nicht der Pi. Spürbar langsamer ist nur **das Zoomen der Weltkarte**: eine Rastung des Mausekursors kostet dort rund eine halbe Sekunde gegen eine Zehntelsekunde unter Windows. Schieben kostet auf beiden gleich viel.

2.5 Start aus dem Quellcode

```
python -m venv .venv
.venv/Scripts/pip install -e ".[dev]"
```

Die Oberfläche einmalig bauen:

```
npm --prefix frontend install
npm --prefix frontend run build
```

Danach starten:

```
.venv/Scripts/python -m logrunner
```

Alternativ steht nach der Installation das Kommando `logrunner` zur Verfügung. Ist kein Frontend-Build vorhanden, verbindet sich das Fenster mit dem Vite-Entwicklungsserver auf `http://localhost:5173`.

Eine Windows-Beta baut:

```
.venv/Scripts/python tools/build_release.py
```

2.6 Der Datenordner

Beim ersten Start legt LogRunner sein Datenverzeichnis an. Voreingestellt ist `~/LogRunner/` (unter Windows also `C:\Users\<Name>\LogRunner\`). Die Umgebungsvariable `LOGRUNNER_HOME` überschreibt diesen Ort – so lässt sich das Logbuch etwa auf einen Stick legen.

```

~/LogRunner/
├─ logrunner.sqlite    Das Logbuch selbst
├─ config.toml         Alle Einstellungen
├─ license.json        Vermerk über die angenommene Lizenz
├─ .logrunner.lock     Riegel gegen einen zweiten Start
├─ data/
│   ├─ cty.dat         Länderkennerdatei (DXCC-Auflösung)
│   ├─ solar.json      Zuletzt geholte Zahlen zur Sonnenaktivität
│   ├─ muf.json        Zuletzt geholte MUF-Konturen und Stationswerte
│   ├─ dxcluster.sqlite Meldungshistorie des DX-Clusters
│   ├─ propagation.sqlite Messwerte für die Trendanzeige
│   └─ webview/        Cookies/Anmeldungen der Anzeigebibliothek
├─ backups/            Automatische Sicherungen (die letzten 20)
├─ exports/            Geschriebene ADIF- und Cabrillo-Dateien
├─ logs/               logrunner.log, start.log
├─ shortcuts/          Verknüpfungen der Begleitfenster
└─ clients/            Datenverzeichnisse der Begleitfenster

```

2.7 Das Startfenster und die Lizenz

Vor dem Hauptfenster erscheint ein kleines Startfenster mit dem Logo, der Versionsnummer und dem Hinweis auf den Endnutzer-Lizenzvertrag. Im Normalbetrieb steht es nur die ein, zwei Sekunden, die das Hauptfenster zum Aufbau braucht, und verschwindet dann von selbst.

Beim allerersten Start und nach jeder neuen Version ist es dagegen eine Frage: es zeigt den vollständigen Lizenztext und wartet auf eine Antwort.

- **Annehmen und starten** – erst das Kreuz in *Ich habe die Lizenzbedingungen gelesen und nehme sie an*, dann der Knopf. Danach geht das Hauptfenster auf.
- **Abbrechen** (oder das Kreuz der Titelleiste, oder `Esc`) – das Programm beendet sich, ohne das Hauptfenster zu zeigen. Es wird nichts ins Netz geschickt, kein Funkgerät angesprochen und kein Abgleich angestoßen.

Die Zustimmung wird in `~/LogRunner/license.json` vermerkt – mit Version und einem Fingerabdruck des Lizenztextes. Gefragt wird deshalb wieder, sobald eine neue Programmversion läuft **oder** der Lizenztext selbst ein anderer geworden ist. Der Vermerk gilt für die ganze Installation: ein Begleitfenster (`--client`) fragt nicht noch einmal.

Der Lizenztext liegt als `LICENSE.txt` neben `LogRunner.exe`, die Bedingungen der enthaltenen Fremdbestandteile als `THIRD-PARTY-NOTICES.txt` daneben. Beide lassen sich jederzeit ohne laufendes Programm lesen; das Startfenster nennt ihre Pfade.

Der Lizenzvertrag ist nur auf Deutsch verbindlich. Die Beschriftungen des Startfensters folgen der in `config.toml` eingestellten Sprache; in den übrigen fünf Sprachen steht ein entsprechender Hinweis dabei.

3. Benutzeroberfläche & Navigation

3.1 Das Hauptfenster

Das Fenster ist in drei waagerechte Bereiche geteilt: die **Kopfzeile** mit allen Menüs, die **Arbeitsfläche** mit Eingabemaske und Reiterbereich und die **Statusleiste** am unteren Rand.

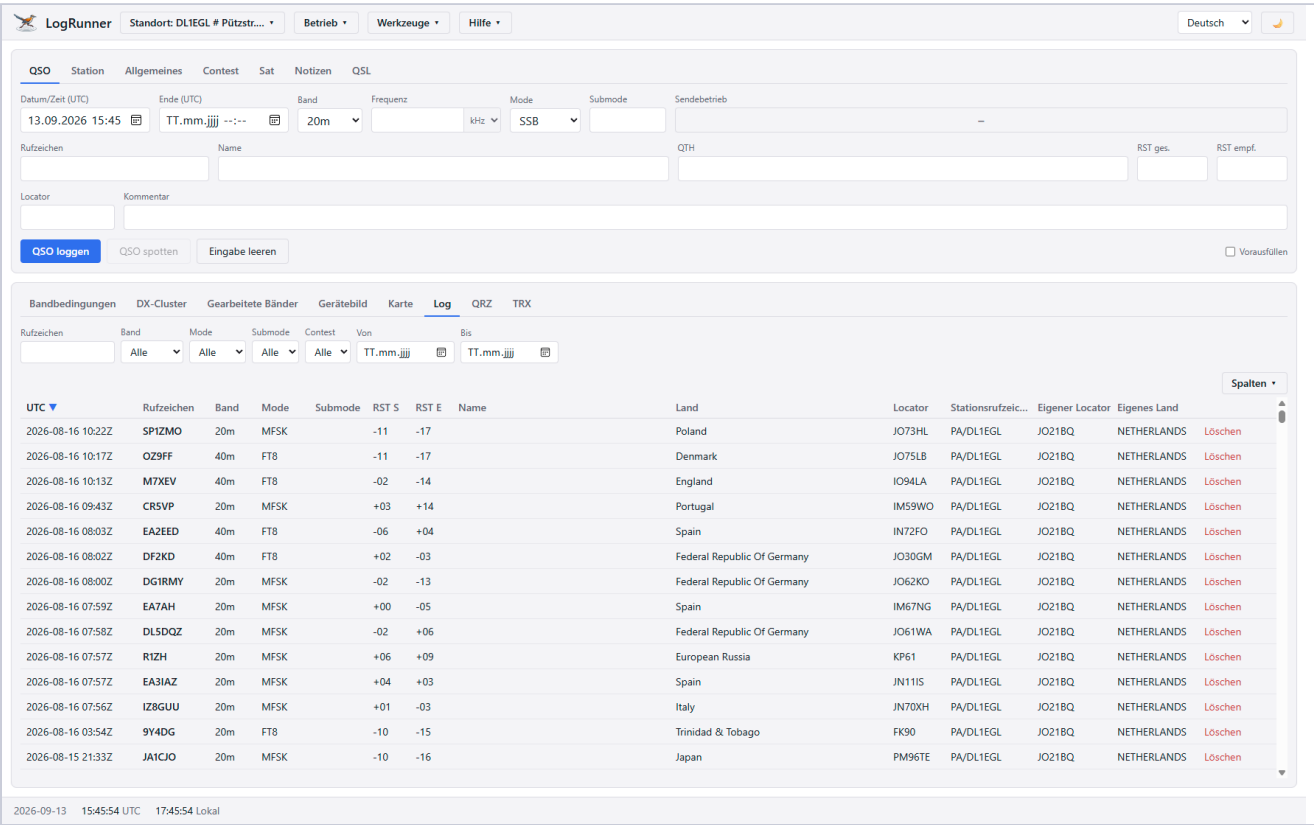


Abb. 1: Das Hauptfenster in der vollständigen Ansicht mit Eingabemaske, Reiterleiste und Log-Tabelle

Die Bilder in dieser Anleitung zeigen das **helle Thema**. Wer das dunkle eingestellt hat, sieht dieselben Bedienelemente an denselben Stellen – der Umschalter ☀️ / 🌙 steht rechts oben in der Kopfzeile.

Kopfzeile (Menüleiste)

Von links nach rechts:

Element	Bedeutung
LogRunner	Programmname
Standort ▾	Der Stationsstandort, unter dem das nächste QSO geloggt wird. Der Knopf nennt den aktiven Standort samt Rufzeichen; ein Haken markiert ihn in der Liste.
Betrieb ▾	Logbuch (volle Maske) oder Contest (die kurze Contest-Zeile).
Preset ▾	Die Fensterkombinationen der LiveLink. Ein Klick fährt das Preset hoch und schließt die Begleitfenster, die nicht dazugehören. Der Knopf nennt das Preset, das gerade läuft; im Menü stehen die Namen, einer davon mit Haken. Steht nur da, wenn die LiveLink eingeschaltet ist und dieses Fenster den Hub führt – ein Client hat keine eigenen Presets und könnte sie auch nicht starten. Angelegt werden sie unter <i>Werkzeuge ▸ LiveLink ▸ Begleitfenster mit starten</i> .

Die ganze Kopfzeile lässt sich abschalten, und der Weg dahin ist der **Rechtsklick** irgendwo im Fenster: *Menüband aus*, und wenn es weg ist, steht an derselben Stelle *Menüband ein*. Das geht in jeder Ansicht, auch in der kompakten, und auch über einem Eingabefeld.

Es ist der einzige Weg, und das mit Absicht: das Werkzeugmenü verschwindet mit der Kopfzeile, ein Haken darin wäre also nur so lange erreichbar, wie man ihn nicht gebraucht hat. Gedacht ist die Einstellung für ein Begleitfenster, das nur eine Karte oder das Gerätebild zeigt und keins dieser Menüs braucht.

Das Menü *Werkzeuge*

Die Einträge stehen alphabetisch nach ihrer übersetzten Bezeichnung, sortieren sich in jeder Sprache also anders. Nur *Alle QSOs löschen* ist ausgenommen: es steht unten hinter einem Trennstrich und in Rot.

Eintrag	Zweck
 ADIF-Import/-Export	QSOs aus <code>.adi</code> -Dateien einlesen und herauschreiben
 Anzeige	QRZ-Öffnungsart, IARU-Bandplanregion
 Bildaufnahme	Capture-Karte einrichten, Bild im Netz anbieten
 Conteste	Contestliste und Cabrillo-Einreichung
 DX-Cluster	Knoten, Login-Rufzeichen, Puffergröße, Skimmer – auf einem Client der Haken <i>Hub folgen</i>
 LiveLink	Hub/Client, Maske, Begleitfenster-Verknüpfungen
 Rufzeichen-Abfrage	QRZ.com, HamQTH, callook.info
 Stationsstandorte	Eigene Standorte samt Stammdaten anlegen
 Statistik	Auswertungen des Logs
 Ticker	Stationsuhr, Rufzeichen und QSO-Zähler für einen eigenen Schirm
 TRX-Steuerung	Anbindung an den Transceiver
 Vorgaben für QSOs	Benannte Sätze von Startwerten für die Maske
 Wavelog-Abgleich	Adresse, Schlüssel, Hochladen und Abholen
 WSJT-X-Übernahme	UDP-Empfang von WSJT-X
 xOTA/SCP-Listen	POTA-, SOTA-, IOTA- und SCP-Listen holen
 Alle QSOs löschen	Zerstörend – siehe Warnung in Abschnitt 4.9

Jeder Eintrag öffnet einen Dialog über der Arbeitsfläche. **Esc** oder der Knopf *Schließen* schließt ihn; der Cursor springt danach zurück in das Rufzeichenfeld.

Die Eingabemaske

Über der Maske stehen sieben Reiter. Ein Reiter, in dem etwas eingetragen ist, wird markiert – so bleibt nichts verborgen, was ein Import oder ein Abruf mitgebracht hat.

Reiter	Inhalt
QSO	Die eigentliche Verbindung – siehe Zeilenaufteilung unten
Station	Empfangsband und -frequenz (Cross-Band, Satellit), Operator, TX-Leistung
Allgemeines	DXCC, Land, Kontinent, Region, CQ-/ITU-Zone, Bundesstaat, Landkreis, Ausbreitungsart, Antennenweg, E-Mail, IOTA, SOTA, WWFF, POTA, SIG, DOK
Contest	Contest-Kürzel, beide laufenden Nummern, beide Austauschfelder
Sat	Satellitenname, Satellitenmode, Antennen-Azimut und -Elevation

Reiter	Inhalt
Notizen	Mehrzeilige QSO-Notiz (ADIF NOTES)
QSL	Sechs Wege, je <i>Gesendet</i> und <i>Erhalten</i> : QSL-Karte, eQSL, LoTW, QRZ, Clublog, DCL – siehe unten

Der Reiter **QSO** ist in drei Zeilen geordnet:

Betriebsart und Unterart. Die Auswahl zeigt oben die neun, die man im Betrieb anfasst, darunter *weitere ADIF-Betriebsarten* mit allen übrigen. *FT4*, *JS8* und *PSK31* sind in ADIF Unterarten – gewählt wird das gewohnte Wort, ins Log geht das Paar (FT4 wird zu MFSK mit Unterart FT4). Der Bandfilter findet ein QSO unter beidem.

1. **Wann und womit:** Datum/Zeit (UTC) · Ende (UTC) · Band · Frequenz · Mode · Submode · *Sendebetrieb* (die On-Air-Anzeige).
2. **Die Gegenstation:** Rufzeichen · Name · QTH · RST ges. · RST empf.
3. **Ergänzungen:** Locator · Kommentar.

Darunter die Knopfleiste: **QSO loggen** · **QSO spotten** · **Eingabe leeren** · (bei laufendem Contest) **Contest-Zeile / Volle Maske**. Ganz rechts steht der Haken **Vorausfüllen**. Zwischen den Knöpfen erscheinen bei Bedarf Hinweisplaketten: *gearbeitet: n QSOs auf m Bändern*, die Entfernung samt Peilung, die laufende Contestnummer und die rote Dupe-Warnung.

Abb. 3: Die Eingabemaske im Reiter QSO mit den drei Feldzeilen und der Knopfleiste

Der Reiter QSL

Sechs Wege, eine QSL loszuwerden oder zu bekommen, und jeder steht in einem eigenen Abschnitt mit Überschrift:

Abschnitt	Felder
QSL-Karte	Gesendet, Sendemethode, Gesendet via, Erhalten, Empfangsmethode, QSL-Nachricht Gesendet, QSL-Nachricht Erhalten
eQSL	Gesendet, Erhalten
LoTW	Gesendet, Erhalten
QRZ	Gesendet, Erhalten
Clublog	nur Gesendet – ADIF kennt bei Club Log keine Empfangsrichtung
DCL	Gesendet, Erhalten

Die fünf elektronischen Wege stehen **nebeneinander**, einer je Spalte, mit ihren beiden Feldern darunter. Ab etwa 1000 px Fensterbreite passen alle fünf in eine Zeile; wird das Fenster schmaler, brechen sie auf drei und dann auf zwei Spalten um.

Die beiden **Nachrichtenfelder** stehen bei der Karte und nicht bei eQSL: ADIF führt `QSLMSG` und `QSLMSG_RCVD` als die Nachricht der *Karte*, und eine eQSL-eigene gibt es im Standard nicht.

Die Stellungen sind **ADIF-Buchstaben und keine Häkchen** – *Angefordert* und *Verifiziert* überstehen so eine Bearbeitung, statt auf ja/nein zusammenzufallen. QRZ und Clublog haben eine Stellung mehr als die anderen:

Geändert – Upload steht aus.

Zwei Stellungen bietet die Maske **nicht mehr** zur Auswahl an, weil ADIF sie nur zum *Einlesen* vorsieht: *Verifiziert* bei „Erhalten“ und *Manager* bei den beiden Wegen. Ein QSO, das sie aus einem Import mitgebracht hat, behält sie trotzdem und zeigt sie an.

Leer ist nicht dasselbe wie Nein. Bei eQSL, LoTW, QRZ, Clublog und DCL steht ab Werk — *keine Angabe* —. LogRunner lädt zu keinem dieser Dienste selbst etwas hoch; ein *Nein* auf jedem QSO wäre eine Auskunft über etwas, das nie jemand gefragt hat. *Nein* heißt **nicht gesendet**, leer heißt **nichts bekannt**. Nur *LoTW erhalten* hat die leere Stellung nicht – das Feld gibt es seit der ersten Fassung mit *Nein*.

Die Datumsfelder stehen nicht in der Maske. Zu jedem der sechs Wege gibt es in ADIF ein Datum für das Senden und eines für das Empfangen; sie kommen mit einem **ADIF-Import** herein (aus Wavelog, von QRZ.com, aus Club Log) und lassen sich als **Spalten der Log-Tabelle** einblenden – unter *Spalten* ▾, in der Gruppe *QSL*. Von Hand einzutragen wären sie eine Zeile Tipparbeit für etwas, das ohnehin ein anderes Programm entscheidet. Eine Bearbeitung in der Maske lässt sie unangetastet.

Der Reiterbereich

Unter der Maske liegt eine Reiterleiste. Auch sie ist alphabetisch nach den übersetzten Namen sortiert.

Reiter	Inhalt
Bandbedingungen	Sonnenfluss, Sonnenflecken, A-/K-Index, MUF, Röntgenfluss, Störpegel und die Einschätzung je Bandgruppe, dazu Trendpfeile, der Pfadindikator für eine Strecke und der Bandindikator für die Kontinente
DX-Cluster	Meldungen als Tabelle, Band-Map oder Bandaktivität (<i>nur bei eingeschaltetem Cluster</i>)
Gearbeitete Bänder	Bandmatrix zum Rufzeichen in der Maske
Gerätebild	Der Schirm des Transceivers über die Capture-Karte
Karte	Weltkarte, Globus oder Azimutkarte
Log	Filterzeile und Log-Tabelle (<i>beim Start vorgewählt</i>)
QRZ	Was die Rufzeichendienste zur Station in der Maske wissen
TRX	Spektrum aus dem Bandscope und die Bedienleiste des Geräts (<i>nur wenn eingeschaltet</i>)

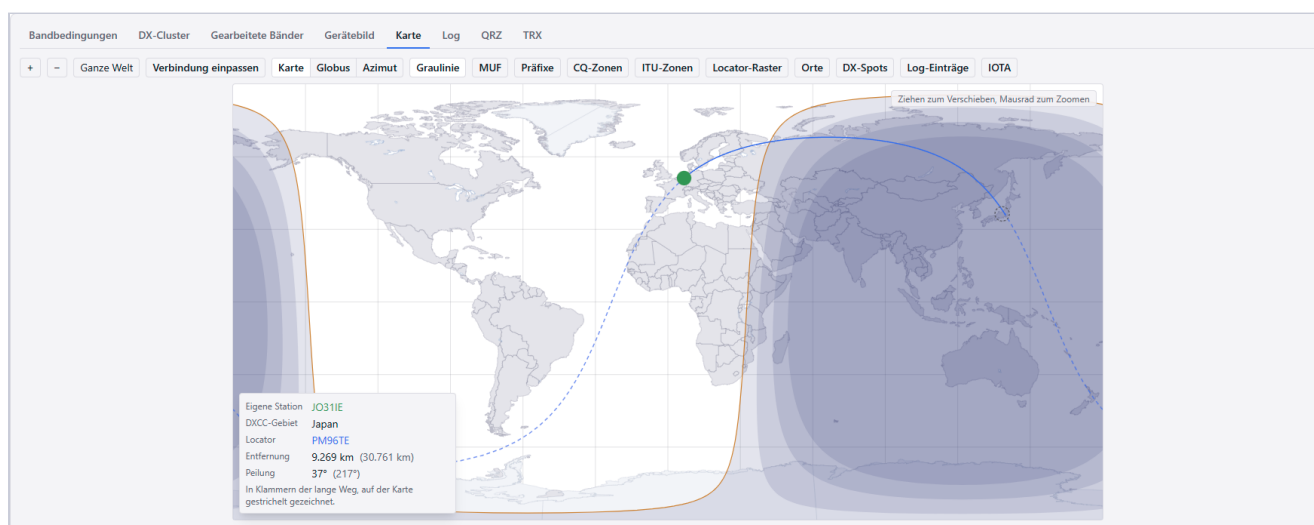


Abb. 4: Der Reiterbereich mit der geöffneten Weltkarte

Die Statusleiste

Am unteren Rand stehen dauerhaft: **Datum**, **UTC-Zeit** und **Ortszeit**. Ist die jeweilige Funktion eingeschaltet, kommen Lämpchen dazu – der Zustand der WSJT-X-Übernahme, der TRX-Steuerung (mit rotem **On Air**, solange gesendet wird) und der LiveLink (mit Rolle, **Namen dieses Fensters** und Zahl der verbundenen PCs).

Der Name ist der, unter dem das Fenster in der PC-Liste und in der Bandzuweisung steht – der vom Hub zugeteilte also, der eine Ziffer trägt, wenn der Wunschname schon vergeben war. Er steht dort, weil die Rolle allein die Frage nur halb beantwortet: stehen mehrere Begleitfenster derselben Bauart offen – zwei Transceiver-Fenster etwa –, sagt „Client“ an jedem von ihnen dasselbe.

Die Leiste lässt sich unter *Werkzeuge* ▶ *Anzeige* abschalten.

3.2 Die Ansicht

Sie steht im Rechtsklick-Menü, an jeder Stelle des Fensters – nicht in der Kopfzeile. Was zur Wahl steht, richtet sich danach, was dieses Fenster zeigt (Maske, Abschnitt 3.1 und *Werkzeuge* ▶ *LiveLink*):

Das Fenster zeigt	Zur Wahl stehen
Alles (ganzes Fenster)	<i>Kompakt</i> · <i>Vollständig</i>
TRX (--mask scope)	<i>Wasserfall</i> · <i>Gerät bedienen</i> · <i>Vollständig</i>
Bandbedingungen (--mask solar)	<i>Sonnenaktivität</i> · <i>Pfadindikator</i> · <i>Bandindikator</i> · <i>Detail</i>
Karte, Log, DX-Cluster, Gerätebild, ...	– nichts; dort gibt es nur diesen einen Bereich

Ein Haken zeigt, welche gerade gilt. Die Wahl gilt nur für *dieses* Fenster, steht in dessen `config.toml` und übersteht einen Neustart.

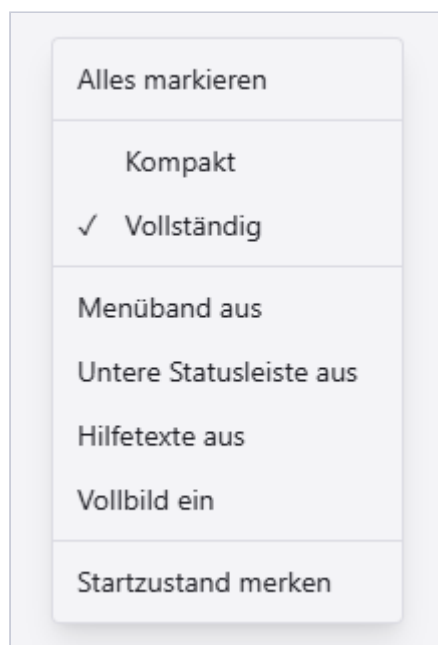


Abb. 5: Das Rechtsklick-Menü im ganzen Fenster: Alles markieren, Kompakt und Vollständig, darunter Menüband, Statusleiste, Hilfetexte, Vollbild und Startzustand merken

Unter der Ansicht stehen im selben Menü die Schalter für das **Menüband**, die **untere Statusleiste**, die **Hilfetexte** und das **Vollbild**, ganz unten *Startzustand merken* (Abschnitt 4.2).

Alles (ganzes Fenster)

- **Vollständig** zeigt die ganze Fläche und passt sich so an, dass alles ohne Scrollen in die Fensterhöhe passt.
- **Kompakt** lässt nur die Eingabemaske stehen und zieht das Fenster auf deren Maß zusammen – für Contestbetrieb oder für ein Fenster, das neben einem Wasserfall in der Ecke parkt.

Beide setzen das Fenster auf die Breite, in der die Maske so steht, wie sie gezeichnet ist.

Abb. 6: Die kompakte Ansicht: nur die Eingabemaske

TRX

Auf einem Rechner, der allein den TRX-Bereich zeigt, gibt es keine Eingabemaske, die man weglassen könnte – die Frage geht deshalb eine Ebene tiefer und betrifft die zwei Hälften dieses Bereichs:

- **Wasserfall** – nur die Anzeige: Kurve, Wasserfall und die DX-Meldungen darüber. Ein reines Wasserfall-Fenster für den Bildschirm nebenan.
- **Gerät bedienen** – nur die Reglerleiste, aufgeklappt und ohne den Klapp-Knopf davor. Solange sie allein steht, fragt LogRunner das Gerät **nicht** mehr nach Spektren ab: die Sweeps laufen über dieselbe CI-V-Leitung wie die Regler, zehnmal je Sekunde, und wer diese Ansicht wählt, hat die Leitung für die Regler bestellt.
- **Vollständig** – beides, so wie der Reiter *TRX* im vollen Fenster aussieht.

Der Wasserfall behält seine Geschichte nicht, wenn man ihn weglegt: sie ist die Zeichnung selbst. Nach *Gerät bedienen* und zurück fängt das Bild neu an.

Bandbedingungen

Dasselbe eine Ebene tiefer, auf einem Rechner, der allein diesen Bereich zeigt. Er besteht aus drei Teilen, und jeder von ihnen ist für sich ein Fenster wert:

- **Sonnenaktivität** – die Zahlen aus dem Feed: die acht Karten mit ihren Verlaufslinien, die Bandtabelle für Tag und Nacht und die VHF-Zeile. Mit ihnen die Kopfzeile, in der *Aktualisieren* und der Schalter stehen.
- **Pfadindikator** – nur die eine Strecke: das Eingabefeld für Rufzeichen oder Präfix und die Bänderreihe darunter.
- **Bandindikator** – nur das Kontinentraster, das an keinem Rufzeichen hängt. Die Ansicht für den zweiten Schirm: *wo komme ich gerade überhaupt hin*.
- **Detail** – alles, so wie der Reiter *Bandbedingungen* im vollen Fenster aussieht.

Das Fenster misst nach jeder Wahl seine Höhe nach und zieht sich auf das zusammen, was noch dasteht.

Die Quellenzeile gehört zu den Zahlen und steht deshalb nur bei *Sonnenaktivität* und *Detail*: sie nennt, wessen Tabelle das ist. Woher die beiden Indikatoren ihre Daten haben, sagt die Legende des MUF-Layers auf der Karte.

4. Funktionsbeschreibung & Workflows

4.1 Workflow 1 – Ein QSO loggen

Das ist die Hauptfunktion des Programms.

Vorbereitung (einmalig). Unter *Werkzeuge* ▶ *Stationsstandorte* mindestens einen Standort anlegen: Bezeichnung, eigenes Rufzeichen, Locator, QTH und was sonst dazugehört. Der Knopf *Aus Rufzeichen ergänzen* füllt Land und Zonen aus `cty.dat`. Der mit ★ markierte Standort ist aktiv.

⚠ **Achtung:** Die Standortangaben werden beim Loggen in **jedes** QSO hineinkopiert – und zwar der Standort zum Zeitpunkt des QSOs. Wer später umzieht und den Standort ändert, ändert damit nicht rückwirkend, wo alte QSOs gemacht wurden. Das ist beabsichtigt.

Ablauf je Verbindung:

1. **Rufzeichen tippen.** Ab dem ersten Zeichen geschieht mehreres zugleich: * Ein Vorschlagsfeld öffnet sich mit Rufzeichen aus dem eigenen Log (hervorgehoben) und aus der Super-Check-Partial-Liste. * Die Log-Tabelle unten filtert sich auf das Getippte. * Die Uhr für *Ende (UTC)* beginnt mitzulaufen. * Steht das Rufzeichen schon im Log, erscheint die Plakette „gearbeitet: n QSOs auf m Bändern“. * Ein **Doppelklick** im Feld öffnet das QRZ.com-Profil.
2. **Band, Frequenz und Betriebsart** setzen – von Hand, aus einer angeklickten DX-Meldung oder automatisch vom Transceiver, wenn die TRX-Steuerung läuft.
3. **Rapporte, Name, QTH, Locator, Kommentar** eintragen, soweit gegeben. Sobald ein Locator dasteht und der eigene Standort einen hat, erscheint die Entfernung samt Peilung – kurzer Weg, in Klammern der lange.
4. **QSO loggen** drücken (oder Enter). Dabei geschieht automatisch: * Land, Kontinent, CQ-/ITU-Zone und die DXCC-Nummer werden aus `cty.dat` ergänzt, sofern die Felder leer sind. * Die Stammdaten des aktiven Standorts werden aufgeprägt. * Läuft ein Contest, bekommt das QSO dessen Kürzel, die nächste laufende Nummer und den festen Austausch. * Das QSO wird in die Wavelog-Warteschlange gestellt – unabhängig davon, ob der Abgleich überhaupt eingerichtet ist. * Die Maske wird geleert; mit gesetztem Haken *Vorausfüllen* bleiben Band, Mode, Submode und Frequenz stehen.

Ein QSO ändern. Ein Klick auf eine Zeile der Log-Tabelle lädt das QSO in die Maske; der Knopf heißt dann *Speichern*, daneben steht *Abbrechen*. Ein Doppelklick auf die Zeile öffnet stattdessen QRZ.com.

Ein QSO löschen. Über das Aktionsfeld der Tabellenzeile, mit Rückfrage.

⚠ **Achtung:** Ein gelöscht QSO ist in LogRunner nicht wiederherstellbar. Wurde es bereits zu Wavelog hochgeladen, bleibt es dort stehen – das Löschen wirkt nur lokal.

Suchmuster statt Rufzeichen. Im Rufzeichenfeld gelten ? für genau ein und * für beliebig viele Zeichen (??1EGL findet jedes sechsstellige Rufzeichen, das auf 1EGL endet). Solange ein Jokerzeichen im Feld steht, verweigert LogRunner das Loggen und sagt das: „Nicht geloggt – im Rufzeichenfeld steht ein Suchmuster“. Ohne Joker gilt das Getippte als Anfang eines Rufzeichens.

Standort: DL1EGL # Pützstr...
Betrieb
Werkzeuge
Hilfe
Deutsch

QSO

Station

Allgemeines

Contest

Sat

Notizen

QSL

Datum/Zeit (UTC)

Ende (UTC)

Band

Frequenz

Mode

Submode

Sendebetrieb

Rufzeichen

Name

QTH

RST ges.

RST empf.

Locator

Kommentar

QSO loggen

QSO spotten

Eingabe leeren

Vorausfüllen

Bandbedingungen

DX-Cluster

Gearbeitete Bänder

Gerätebild

Karte

Log

QRZ

TRX

Rufzeichen

Band

Mode

Submode

Contest

Von

Bis

UTC

Rufzeichen

Band

Mode

Submode

RST S

RST E

Name

Land

Locator

Stationsrufzeic...

Eigener Locator

Spalten

1 UTC

2 Rufzeichen

3 Band

4 Mode

5 Submode

6 RST S

7 RST E

8 Name

9 Land

10 Locator

11 Stationsrufzeichen

12 Eigener Locator

13 Eigenes Land

QSO

✓ UTC

Standardspalten

NETHERLANDS Löschen

NETHERLANDS Löschen

UTC	Rufzeichen	Band	Mode	Submode	RST S	RST E	Name	Land	Locator	Stationsrufzeic...	Eigener Locator
2026-08-16 10:22Z	SP1ZMO	20m	MFSK		-11	-17		Poland	JO73HL	PA/DL1EGL	JO21BQ
2026-08-16 10:17Z	OZ9FF	40m	FT8		-11	-17		Denmark	JO75LB	PA/DL1EGL	JO21BQ
2026-08-16 10:13Z	M7XEV	40m	FT8		-02	-14		England	IO94LA	PA/DL1EGL	JO21BQ
2026-08-16 09:43Z	CR5VP	20m	MFSK		+03	+14		Portugal	IM59WO	PA/DL1EGL	JO21BQ
2026-08-16 08:03Z	EA2EED	40m	FT8		-06	+04		Spain	IN72FO	PA/DL1EGL	JO21BQ
2026-08-16 08:02Z	DF2KD	40m	FT8		+02	-03		Federal Republic Of Germany	JO30GM	PA/DL1EGL	JO21BQ
2026-08-16 08:00Z	DG1RMV	20m	MFSK		-02	-13		Federal Republic Of Germany	JO62KO	PA/DL1EGL	JO21BQ
2026-08-16 07:59Z	EA7AH	20m	MFSK		+00	-05		Spain	IM67NG	PA/DL1EGL	JO21BQ
2026-08-16 07:58Z	DL5DQZ	20m	MFSK		-02	+06		Federal Republic Of Germany	JO61WA	PA/DL1EGL	JO21BQ
2026-08-16 07:57Z	R1ZH	20m	MFSK		+06	+09		European Russia	KP61	PA/DL1EGL	JO21BQ
2026-08-16 07:57Z	EA3IAZ	20m	MFSK		+04	+03		Spain	JN11IS	PA/DL1EGL	JO21BQ
2026-08-16 07:56Z	I28GUU	20m	MFSK		+01	-03		Italy	JN70XH	PA/DL1EGL	JO21BQ
2026-08-16 03:54Z	9Y4DG	20m	FT8		-10	-15		Trinidad & Tobago	FK90	PA/DL1EGL	JO21BQ
2026-08-15 21:33Z	JAT1CO	20m	MFSK		-10	-16		Japan	PM96TE	PA/DL1EGL	JO21BQ

2026-09-13 15:49:29 UTC 17:49:29 Lokal

Abb. 7: Die Log-Tabelle mit Filterzeile und Spaltenauswahl

4.2 Workflow 2 – Datenaustausch, Import, Export, Schnittstellen

ADIF-Import und -Export

Werkzeuge ► ADIF-Import/-Export

Import: ADIF importieren ... öffnet den Dateiauswahldialog (.adi , .adif). Vorher ist zu wählen, was bei Dubletten geschehen soll:

Regel	Wirkung
Überspringen	Vorhandene QSOs bleiben unangetastet
Aktualisieren	Der vorhandene Datensatz wird mit den Werten aus der Datei überschrieben
Trotzdem anlegen	Der Datensatz wird ein zweites Mal angelegt

Groß- und Kleinschreibung spielt bei Rufzeichen keine Rolle. DL1EGL , D11eg1 , d11eg1 und d11eG1 sind dasselbe Rufzeichen; der Import erkennt sie als dieselbe Station und schreibt sie in Großbuchstaben ins Logbuch – das Rufzeichen des Gegenübers ebenso wie *Stationsrufzeichen* und *Operator*. Das Feld *QSL via* bleibt unangetastet: dort steht in der Praxis Freitext (*via bureau*, *eQSL*, eine Mailadresse), und den würde Großschreiben verderben.

Der Bericht nennt danach: importiert, aktualisiert, übersprungen, fehlerhafte Datensätze und den Pfad der automatisch angelegten Sicherung. Felder, die LogRunner nicht kennt, gehen nicht verloren – sie werden in `qso_extra` aufbewahrt und beim Export wieder mitgeschrieben. Importierte QSOs werden dem aktiven Standort zugeordnet und in die Wavelog-Warteschlange gestellt.

Achtung: Vor **jedem** Import wird automatisch eine vollständige Sicherung der Datenbank in `~/LogRunner/backups/` angelegt. Ein Import mit der Regel *Aktualisieren* überschreibt bestehende QSOs – der Weg zurück führt nur über diese Sicherung.

Export: ADIF exportieren schreibt nach `~/LogRunner/exports/` in eine Datei `logrunner-<JJJJMMTT-HHMMSS>.adi`. Mit dem Haken *Nur die gefilterten QSOs exportieren* gilt der Filter der Log-Tabelle – also Rufzeichen, Band, Mode, Submode, Contest und Zeitraum.

Wavelog-/Cloudlog-Abgleich

Werkzeuge ▶ Wavelog-Abgleich

1. **Wavelog-Adresse** eintragen – nur die Adresse des Logbuchs, ohne `/api/...`; ein mitkopierter Endpunktpfad wird beim Speichern entfernt.
2. **API-Schlüssel** eintragen. Die API-Generation ergibt sich aus dem Schlüssel selbst: Tokens mit dem Präfix `wl2_` nutzen **API v2**, alles andere die **API v1**. Für v2 braucht das Token mindestens die Scopes `qso:read` und `qso:write`.
3. **Verbindung prüfen** – meldet die erkannte Wavelog-Version und, bei v2, fehlende Scopes.
4. **Station** wählen (die Wavelog-Standortnummer).

Knopf	Wirkung
Jetzt hochladen	Schickt alle wartenden QSOs
Jetzt abholen	Holt neue QSOs von Wavelog, je Standort inkrementell
Änderungen senden	Überschreibt in Wavelog QSOs, die hier nach dem Hochladen geändert wurden (nur API v2)

Steht etwas an, listet eine Tabelle unter den Knöpfen die **wartenden QSOs** auf – Zeit, Rufzeichen, Band, Frequenz, Betriebsart und beide Rapporte. Die Spalte *Art* sagt, welchen Weg jedes nimmt: *Neu* geht mit *Jetzt hochladen* hinaus, *Geändert* mit *Änderungen senden*. Ein Δ davor heißt, dass der letzte Versuch fehlschlug; das QSO bleibt in der Warteschlange und geht beim nächsten Mal wieder mit. Die Tabelle ist nur zum Nachsehen – angehakt oder ausgewählt wird darin nichts. Bei sehr vielen wartenden QSOs zeigt sie die ersten 200 und sagt darunter, wie viele nicht aufgeführt sind.

Ohne Verbindung bleiben die QSOs in der Warteschlange und gehen beim nächsten Versuch sowie beim nächsten Programmstart hinaus. Die Zahl der Wartenden steht in der Kopfzeile.

Unter *Stationsstandorte* lässt sich jedem lokalen Standort ein eigenes Wavelog-Logbuch zuordnen; der Knopf *Aus Wavelog holen* übernimmt die Standorte des Kontos – als Vorschau, die Feld für Feld zeigt, was sie ändern würde.

⚠ Achtung: Der API-Schlüssel steht im Klartext in `config.toml`. Liegt der Datenordner auf einem USB-Stick oder in einem Cloud-Ordner, kann jeder ihn lesen, der Zugriff darauf hat.

Wavelog-Abgleich

Schließen

☒ Wavelog-Abgleich aktiv

Wavelog-Adresse

https://log.dclnext.darc.de

API-Schlüssel API v2

Station

6465

Verbindung prüfen

Speichern

Nur die Adresse des Logbuchs, ohne `/api/...` – ein mitkopierter Endpunktpfad wird beim Speichern entfernt.

Δ Der API-Schlüssel steht im Klartext in `config.toml`. Diese Datei liegt im LogRunner-Ordner – auf einem USB-Stick oder in einem Cloud-Ordner ist sie für jeden lesbar, der Zugriff darauf hat.

Die API-Generation ergibt sich aus dem Schlüssel: Tokens mit dem Präfix `wl2_` nutzen API v2, alles andere die Legacy-API v1. Für v2 braucht das Token mindestens die Scopes `qso:read` und `qso:write`.

Jetzt hochladen

Änderungen senden

Jetzt abholen

Wartet auf Upload: 0 · Geändert: 0 · Hochgeladen: 791

Abb. 8: Der Wavelog-Abgleich mit Adresse, Schlüssel und Warteschlange

Cabrillo-Einreichung

Werkzeuge ▶ Conteste ▶ Cabrillo-Einreichung – siehe Abschnitt 4.3.

DX-Cluster (Telnet)

Werkzeuge ▶ DX-Cluster schaltet den Cluster ein und verwaltet die Knoten. Drei sind voreingestellt (DXFun, VE7CC, NC7J), alle drei sind ganz normale, änderbare Zeilen. Jeder Knoten hat Adresse, Port, ein eigenes Login-Rufzeichen und eine Liste von Befehlen, die nach dem Login gesendet werden.

Im Reiter *DX-Cluster* stehen drei Ansichten zur Wahl:

- **Tabelle** – die Meldungen der Reihe nach, mit Rohausgabe und Befehlszeile.
- **Band-Map** – eine senkrechte Skala je Band, eine Zeile je Frequenz, mit der VFO-Marke des eigenen Geräts und Farben nach IARU-Bandplan (CW blau, digital grün, Phonie rot, Baken gelb).
- **Bandaktivität** – wie viele Meldungen je Viertelstunde eingegangen sind, nach Kontinent aufgeteilt.

Ein Klick auf eine Meldung – in der Liste, in der Band-Map, auf der Karte oder im Wasserfall – übernimmt Rufzeichen, Frequenz, Band und, sofern genannt, den Mode in die Eingabemaske; Datum und Uhrzeit bleiben unberührt. Ein **Doppelklick** öffnet zusätzlich QRZ.com. Läuft die TRX-Steuerung mit eingeschaltetem *Angeklickte DX-Meldung an den Transceiver geben*, stellt derselbe Klick das Gerät ein.

Filter. Rufzeichen, kHz von/bis, Band, Mode, Spotter, Region, Spotter-Region und Kommentar. Jokerzeichen: * für beliebig viele, ? für genau ein Zeichen. Ohne Joker gelten Rufzeichen, Spotter, Region, Band und Mode als Anfang, der Kommentar als Teil. Mehrere Werte in einem Feld werden mit Komma oder Semikolon getrennt (FT8,FT4) – nur im Feld *kHz* nicht, dort ist das Komma ein Dezimaltrennzeichen. Filtersätze lassen sich unter einem Namen speichern.

Alarme. Regeln der Form *Feld + Muster* melden sich sichtbar (Zähler am Reiter) und hörbar. Für die Frequenz ist ein Bereich in kHz einzutragen (14000-14350 , 14000- , -7100). Jede Regel kann eine eigene Klangdatei haben; ein Hauptschalter und ein Lautstärkeregler gelten für alle. Geprüft wird jeder **neu** eintreffende Spot, auch wenn das Fenster gerade woanders steht.

Selbst melden. *QSO spotten* in der Eingabemaske füllt das Meldeformular im Cluster-Panel vor.

 **Achtung:** Eine Meldung geht unter dem eigenen Rufzeichen weltweit hinaus und lässt sich nicht zurückholen. Das Formular zeigt vor dem Absenden, wie die Zeile aussehen wird, und warnt bei Selbstmeldungen sowie bei einem Mode, den der Bandplan an dieser Stelle nicht vorsieht.

Skimmer-Meldungen (RBN) sind ab Werk aus. Ein Knoten mit RBN-Anschluss schickt Tausende je Stunde und würde den Puffer (voreingestellt 500 Meldungen) binnen Minuten mit Maschinenmeldungen füllen.

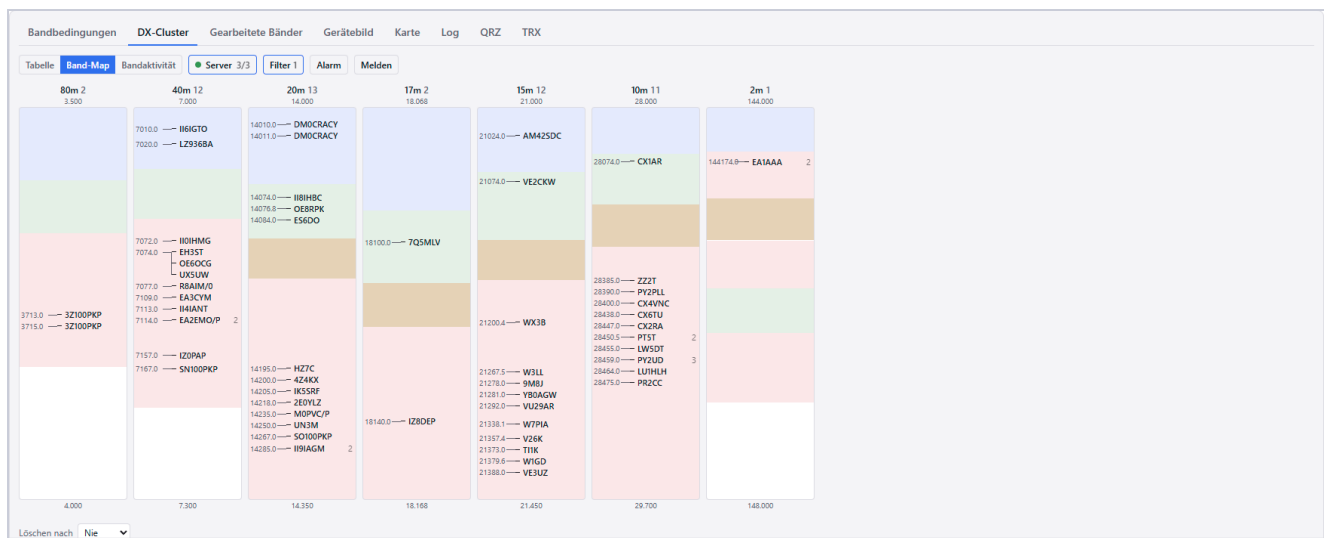


Abb. 9: Die Band-Map des DX-Clusters: eine Spalte je Band, die Meldungen an ihrer Frequenz, Farben nach Bandplan

WSJT-X-Übernahme

Werkzeuge ▶ WSJT-X-Übernahme

Eingeschaltet öffnet LogRunner einen UDP-Port und trägt jede in WSJT-X geloggte Verbindung sofort ins Log ein – unter dem aktiven Stationsstandort, mit Band, Mode, Rapporten, Locator und Zeiten.

In WSJT-X: *Datei* ▶ *Einstellungen* ▶ *Reporting*, „UDP Server“ auf dieselbe Adresse und denselben Port setzen. Ab Werk ist das 127.0.0.1:2237. Läuft daneben JTAAlert oder GridTracker, hört immer nur eines der Programme mit – dann in beiden Programmen dieselbe **Multicast-Adresse** eintragen, etwa 224.0.0.1.

⚠ Achtung: Dies ist die einzige Funktion, die ohne Rückfrage ins Log schreibt. Deshalb ist sie ab Werk aus, und deshalb steht ihr Zustand, solange sie läuft, in der Statusleiste.

TRX-Steuerung

Werkzeuge ▶ TRX-Steuerung

Drei Anbindungen stehen zur Wahl – immer nur eine zur Zeit, denn der Transceiver hat einen Steueranschluss:

Anbindung	Wofür	Einzustellen
Icom CI-V (direkt)	Icom-Geräte an der seriellen Schnittstelle	Gerät, Schnittstelle, Baudrate, CI-V-Adresse (zwei Hexstellen, IC-7300: 94, IC-705: A4)
Hamlib (rigctl)	Alle Geräte, die Hamlib kennt – auch Yaesu, Kenwood, Elecraft, Flex	Rechner und Port des laufenden rigctl (ab Werk 127.0.0.1:4532)
Omnirig	Wenn ein zweites Programm denselben Transceiver braucht	Rig1 oder Rig2

Zwei Schalter bestimmen, was fließt:

- **Änderungen am Gerät in die Maske übernehmen** – wer am VFO dreht, sieht Frequenz, Band, Betriebsart und Submode nachziehen. Ein Feld, in dem gerade der Cursor steht, bleibt unangetastet, ein zur Korrektur geöffnetes QSO ebenfalls.
- **Angeklickte DX-Meldung an den Transceiver geben** – der einzige Weg, auf dem LogRunner das Gerät verstellt.

⚠ **Achtung:** Umgekehrt wird nie gesteuert. Was in die Eingabemaske getippt wird, bleibt in der Eingabemaske. Eine Frequenz, die von Hand entsteht, durchläuft auf dem Weg zu 14.074 erst 1, 14, 140 und 1400 – ein Gerät, das dabei mitlief, stünde die halbe Zeit im Rundfunkband.

Sendezustand. Ist *Sendezustand abfragen* (Befehl *1Ch*) eingeschaltet, zeigen Maske und Statusleiste ein rotes **On Air**, solange gesendet wird. Geräte, die älter sind als etwa 2005, kennen den Befehl nicht; dann sollte die Abfrage aus bleiben.

Wasserfall. *Wasserfall-Spektrum holen* zeigt im Reiter **TRX** das Spektrum des Geräts – 475 Punkte je Durchlauf, je nach Baudrate drei bis zwanzig Durchläufe je Sekunde (siehe unten), mit Kurve, Frequenzachse und VFO-Marke. Ein Klick hinein stimmt das Gerät ab. Läuft der DX-Cluster, stehen dessen Meldungen mit im Bild – die, die in den abgebildeten Ausschnitt fallen.

⚠ **Achtung:** Der Wasserfall geht nur bei Icom-Geräten mit eingebautem Bandscope und nur über deren USB- oder LAN-Anschluss – am [REMOTE] -Anschluss weist das Gerät den Befehl ab. Das Einschalten schaltet außerdem das Bandscope am Gerät selbst ein.

⚠ **Achtung:** Der Wasserfall zeigt nur den Ausschnitt, den das Bandscope gerade abbildet – in **CENTER** ist das die VFO-Frequenz $\pm$ *Span*, in **FIX** der am Gerät eingestellte Randbereich. Meldungen außerhalb dieses Ausschnitts stehen nicht im Bild und lassen sich dort folglich auch nicht anklicken. Bei *Span* ± 5 kHz sind das fast alle: der Ausschnitt ist dann 10 kHz breit, ein Band ist es nicht. Wer die DX-Meldungen im Wasserfall nutzen will, stellt den Span entsprechend weit – oder FIX mit passenden Rändern.

Abfragetakt. Wie oft LogRunner das Gerät nach Frequenz, Betriebsart und Sendezustand fragt. Ab Werk 300 Millisekunden; das ist für die Anzeige schnell genug und lässt der Leitung Luft. Kürzere Werte bringen wenig – die Oberfläche selbst sieht ohnehin zweimal je Sekunde nach –, kosten aber spürbar Übertragungszeit, und die fehlt dann dem Wasserfall.

Baudrate und Wasserfall teilen sich eine Leitung. Ein Durchlauf des Spektrums sind rund 585 Bytes. Bei 19200 Baud dauert das etwa 0,3 Sekunden, also höchstens gut drei Durchläufe je Sekunde; bei 115200 Baud sind rund zwanzig möglich. Wer den Wasserfall flüssiger haben will, stellt also zuerst die Baudrate hoch – im Gerätemenü **und** in LogRunner, beide müssen übereinstimmen. Ein Abfragetakt von 100 ms verbraucht bei 19200 Baud dagegen schon rund ein Fünftel der Leitung, ohne dass die Anzeige davon schneller wird.

Über die Icom Remote Utility. Läuft die Verbindung nicht am Kabel, sondern über den virtuellen COM-Port der **Icom Remote Utility**, gilt alles oben – mit einem Unterschied: die Grenze ist dann nicht mehr die eingetragene Baudrate, sondern das, was der Netzweg zwischen den beiden Programmen trägt. Eine hohe Baudrate bedeutet dort vor allem, dass das Gerät *mehr* Daten schickt, und die müssen alle durch diesen Weg.

Gerät bedienen. Unter dem Wasserfall sitzt eine zugeklappte Leiste mit den Bedienelementen des Transceivers. Aufgeklappt hat sie sieben Ebenen mit Reglern und Haken, die sich wie die Reiter darüber umschalten lassen: *Ton* (Lautstärke, Rauschsperr), *Empfang* (HF-Verstärkung, Vorverstärker, Dämpfung, AGC), *Filter* (Störaustaster, Rauschminderung samt Stärke, Auto-Notch, Notch und Notch-Lage, Filter, PBT innen und außen als Regler mit Mitte), *Mithören* (Mithören und Mithörpegel), *Senden* (Sendeleistung, Tuner, Mikrofonverstärkung, Kompressor, VOX), *CW* (Break-in, Gebetempo) und *VFO* (VFO A, VFO B, A = B, A $\rightleftharpoons$ B, Split, RIT, XIT samt **Ablage**, Knopfsperre). Die Ablage ist ein Regler von -9990 über 0 bis +9990 Hz mit einem Eingabefeld daneben, in das sich der Wert auch eintippen lässt. *VFO A* und *VFO B* rasten ein und zeigen, welcher zuletzt gewählt wurde – welcher am Gerät aktiv ist, meldet der Transceiver über CI-V nicht. RIT und XIT teilen sie sich, wie am Gerät auch.

Hinweis zu *Split*: Steht das Gerät auf **Relaisablage** (DUP– oder DUP+ – auf 2 m und 70 cm stellt die Auto-Repeater-Funktion sie selbst ein), bleibt der Haken *Split* leer, und das ist richtig so: CI-V hält Split und Relaisablage in demselben Befehl, und eine Relaisablage ist kein Split. Bis 0.9.3b56 las LogRunner sie als Split – der Haken setzte sich dann ein paar Sekunden nach dem Abwählen von selbst wieder. Dazu kommen zwei Ebenen mit Tasten: **Mode** schaltet die Betriebsart (LSB, LSB-D, USB, USB-D · AM, FM · CW, RTTY), **Band** springt auf ein Band – auf die Stelle, die zur eingestellten Betriebsart passt, denn CI-V kennt keinen Bandwechsel, nur Frequenzen. In beiden Reihen steht die Taste hervorgehoben da, die gerade gilt.

Die Bandtasten sind die, die das Gerät selbst meldet. Beim Aufklappen der Leiste wird es danach gefragt: erst nach seinen festen Bandkanten, und wenn es die nicht hergibt, nach seinem Bandstapelspeicher. Ein IC-7300MK2 nennt so zwölf Bänder von 160 m bis 4 m – 60 m eingeschlossen, für das er nicht einmal eine eigene Bandtaste hat –, ein IC-9700 nennt 2 m, 70 cm und 23 cm. Die Kanten kommen dabei aus der Regionalfassung des Geräts, nicht aus einer Liste im Programm.

Sagt ein Gerät dazu nichts – ältere Modelle, Hamlib, Omnirig –, steht die volle Liste da wie bisher: 2200 m bis 4 m samt **GEN**. Und wer sie auch an einem Gerät sehen will, das antwortet, setzt unter *Werkzeuge* ▶ *TRX-Steuerung* den Haken **Alle Bänder anzeigen** – gedacht für ein umgebautes Gerät oder eine Sonderfreigabe, von der die Bandkanten nichts wissen. **GEN** kommt nur dazu, wenn das Gerät überhaupt Kurzwelle kann; die Taste stellt 10,000 MHz ein, und auf einem reinen UKW-Gerät wäre das nur eine Fehlermeldung. Auf der VFO-Ebene sitzt rechts ein **Abstimmknopf**, der sich mit der Maus drehen lässt (Schrittweite 10 Hz, 100 Hz oder 1 kHz); darunter lässt sich eine Frequenz auch eintippen und mit Enter übernehmen – mit Punkt sind es Megahertz, ohne Punkt Kilohertz.

Der Knopf **Segmente / Balken** oben rechts schaltet die Instrumente zwischen einer Kette aus Kästchen – wie die Anzeige am Gerät – und einem durchgehenden Balken um; die Wahl bleibt über den Neustart erhalten. Die Instrumente stehen auf jeder Ebene rechts außen untereinander und bleiben beim Umschalten an derselben Stelle; auf *Senden* zeigt die Spalte PO, SWR, ALC, COMP, Vd und Id, sonst zusätzlich das S-Meter. Die Bedienelemente selbst stehen ebenfalls in Spalten – was zusammengehört, steht untereinander. Es gibt die Leiste nur bei **Icom CI-V**, nicht über Hamlib oder Omnirig.

Instrumente. Darüber steht, was das Gerät gerade misst: S-Meter, PO, SWR, ALC, COMP, Vd und Id – als Balken, und wo es eine Kennlinie gibt, mit der gedeuteten Zahl daneben. Alle sieben stehen immer da; was gerade nicht gemessen wird, ist blass und trägt einen Strich statt einer Zahl. Beim Empfang wird nur das S-Meter geholt, beim Senden nur die Sendewerte – alles andere stünde ohnehin auf null und würde nur die Leitung kosten, die der Wasserfall braucht. Gemessen wird nur, solange die Leiste offen ist.

Regler lassen sich mit dem Musrad bedienen: eine Rastung ist ein Prozent, mit gehaltener Umschalttaste fünf. Der Wert geht dabei laufend an das Gerät, nicht erst beim Loslassen – gedrosselt auf höchstens fünf Befehle je Sekunde, damit die Leitung nicht verstopft, und der zuletzt eingestellte Wert kommt in jedem Fall an.

i ALC, COMP, Vd und Id stehen als Prozent des Vollausschlags da. Das Gerät misst Vd in Volt, Id in Ampere und COMP in Dezibel – das sagt auch der Tooltip –, aber die Umrechnung unterscheidet sich von Modell zu Modell, und eine geratene Zahl wäre schlechter als ein ehrlicher Balken.

Drei Dinge dazu:

- **Gelesen wird beim Aufklappen, danach folgt die Leiste dem Gerät.** Wer am Gerät selbst dreht, sieht die Änderung binnen einer halben Minute auch hier – nachgelesen wird ein Wert je Abfrage, damit dem Wasserfall die Leitung bleibt. Der Knopf *Neu lesen* holt alles sofort.
- **Während des Sendens ruht das Nachlesen.** Dann gehören die Abfragen den Instrumenten. Auch auf einer zähen Verbindung – etwa über die Remote Utility – hält sich die Leiste zurück, damit Frequenz und Sendezustand Vorrang behalten.
- **Was das Gerät nicht kennt, erscheint nicht.** Jedes Element wird einmal gefragt; ältere Geräte zeigen deshalb eine kürzere Leiste.

- **Nichts davon sendet.** Der Tuner lässt sich ein- und ausschalten, aber nicht starten – Abstimmen bleibt die Taste am Gerät.

⚠ Achtung: Die Kombination aus hoher Baudrate, eingeschaltetem Wasserfall und kurzem Abfragetakt kann den virtuellen Port überlasten. Das sieht dann nicht wie eine Überlastung aus: die Remote Utility zeigt weiterhin *connected*, während in LogRunner die Frequenzanzeige stehenbleibt. Bei Problemen zuerst den Abfragetakt auf 300 ms stellen, dann den Wasserfall abschalten und beides einzeln wieder zuschalten.

Abbildung folgt: Der Wasserfall mit Spektrumkurve, Frequenzachse und eingeblendeten DX-Meldungen

Rufzeichen-Abfrage

Werkzeuge ▶ Rufzeichen-Abfrage

Drei Dienste, alle ab Werk aus:

Dienst	Konto	Umfang
QRZ.com	Konto nötig; den vollen Datensatz gibt die XML-Schnittstelle nur an Konten mit dem Abonnement <i>XML Logbook Data</i>	weltweit
HamQTH.com	kostenloses Konto nötig	weltweit, liefert auch den DOK
callook.info	kein Konto	nur US-Rufzeichen (K, N, W, AA–AL)

Gefragt wird in dieser Reihenfolge; jeder Dienst wird je Rufzeichen genau einmal gefragt, und jedes Feld füllt der erste, der es beantwortet. Antworten werden lokal zwischengespeichert und stehen danach auch ohne Netz zur Verfügung. Die Ergebnisse landen im Reiter *QRZ* und füllen Name, QTH, Locator, DOK und weiteres in die Maske.

i Wer LogRunner länger benutzt: das Profilbild und die Angaben daneben – Webseite, Lizenzklasse, LoTW/eQSL/QSL-Karte – gibt es erst seit August 2026. Rufzeichen, die vorher nachgeschlagen wurden, liegen ohne sie im Zwischenspeicher, und der Reiter zeigte deshalb Angaben ohne Bild – das Bild erst nach einem Druck auf *Erneut abfragen*. Das erledigt der Reiter jetzt von selbst: er holt einen solchen Datensatz beim ersten Ansehen einmal nach. Währenddessen steht neben dem Rufzeichen *wird abgefragt*; die Angaben sind schon da, das Bild kommt nach. Auf einem Client geschieht das nicht – dort liegt der Zwischenspeicher nicht.

Im Reiter *QRZ* lässt sich Text **mit der Maus markieren und mit Strg+C kopieren** – im übrigen Programm nicht, damit ein Zug über die Band-Map keine blaue Spur quer durch die Panels zieht. Im *QRZ*-Fenster mit der vollständigen Seite geht es überall.

Ein **Rechtsklick** auf markierten Text bietet *Kopieren* an; in einem Eingabefeld stehen dazu *Ausschneiden*, *Einfügen* und *Alles markieren*. Wo es nichts zu tun gibt, bleiben diese vier weg. Strg+C, Strg+X und Strg+V tun unverändert dasselbe.

Durch einen Strich abgesetzt steht darunter die **Ansicht** – zwei oder drei Einträge mit einem Haken beim geltenden, je nachdem, was dieses Fenster zeigt. Was dort zur Wahl steht, sagt Abschnitt 3.2.

Wieder durch einen Strich abgesetzt stehen dann immer drei Schalter – je *aus*, solange die Sache steht, und *ein*, wenn sie weg ist:

Schalter	Wirkung
Menüband	Die Kopfzeile mit allen Menüs (Abschnitt 3.1)

Schalter	Wirkung
Untere Statusleiste	Die Zeile mit Datum, Uhren und Lämpchen
Hilfetexte	Die kurzen Erklärungen unter Haken und Feldern; Kurzhinweise am Mauszeiger bleiben in jedem Fall

Sie stehen an jeder Stelle des Fensters und in jeder Ansicht, auch über Text und in einem Eingabefeld. Deshalb bringt ein Rechtsklick überall ein Menü, auch dort, wo früher keines kam. Jede der drei Einstellungen gilt nur für *dieses* Fenster.

Darunter steht **Vollbild ein** – und solange es an ist, **Vollbild aus**: das Fenster geht auf den ganzen Schirm und wieder zurück. Es lag bis 0.9.3b15 auf **F11**; die Funktionstasten sollen dem Sprach- und dem CW-Keyer gehören, deshalb ist die Taste wieder frei.

Startzustand merken

Der letzte Eintrag, abgesetzt durch einen Strich. Er ist der einzige in diesem Menü, der etwas dauerhaft festhält: alles darüber verstellt dieses Fenster **jetzt**, dieser Eintrag legt fest, womit LogRunner das **nächste Mal** aufgeht.

Bis zum 02.09.2026 geschah das von selbst – beim Klick auf das Kreuz rechts oben wurden Fenstergröße, Fensterposition und die eingestellten Ansichten festgehalten. Das hatte einen Preis: wer sein Fenster nur einmal zur Seite schob, um etwas anderes anzusehen, oder wer eine Ansicht nur ausprobieren wollte, hatte damit den nächsten Start verstellt, ohne es zu wollen. Jetzt gilt: **ausprobieren kostet nichts, gemerkt wird auf Zuruf**.

Der Eintrag schreibt in einem Zug:

Das Fenster	Größe, Position auf dem Bildschirm, maximiert – dazu die Ecke des QRZ-Fensters, falls in dieser Sitzung eines offen war
Die Ansicht	Kompakt oder Vollständig; auf einem Begleitfenster die Wahl innerhalb seines Bereichs
Das Fenstergerüst	Menüband, untere Statusleiste, Hilfetexte
Sprache und Farbschema	dieselben zwei aus dem Menüband
Die Karte	alle Ebenen und die Projektion
Der Rest	Spalten der Log-Tabelle, Bandplan-Region der Band-Map, Eingabemodus (Log oder Contest), Aufteilung des Wasserfalls, Band-Matrix nach Betriebsart

Nicht mitgenommen wird alles, was in ein QSO schreibt oder ins Netz greift: die Frequenzeinheit der Eingabefelder, *Vorbelegen*, der Stationsstandort, die QSO-Vorgaben und sämtliche Einstellungen unter *Werkzeuge* – Wavelog, Abfragedienste, DX-Cluster, LiveLink, CAT, Bildaufnahme. Die stehen wie bisher sofort in der `config.toml`. Der Grund ist einer: eine Frequenzeinheit, die beim Neustart still auf MHz zurückspringt, weil man das Merken vergessen hat, kostet falsche QSOs – und dafür ist kein Bedienkomfort zu haben.

Der Eintrag sagt, was er getan hat: an der Stelle, an der das Menü stand, erscheint kurz **Startzustand gemerkt**. Ließ sich die Datei nicht schreiben, steht dort in Rot, dass es nicht ging.

Auf einem Wayland-Desktop nimmt LogRunner dafür einen Umweg – seit 0.9.3b43, und der Haken dafür steht unter *Werkzeuge* ▶ *Anzeige* ▶ *Fensterstellen*. Ab Werk ist er gesetzt: das Fenster wird dann über **XWayland** gezeichnet, die X11-Schicht, die auf jedem Wayland-Desktop ohnehin mitläuft, und dort kennt es seine Stelle wieder. Beliebig viele Begleitfenster, jedes mit eigener Maske, jedes merkt sich Stelle und Größe – genau wie unter Windows. Am Desktop ist dafür nichts einzustellen.

Der Haken kostet etwas: auf einem Raspberry Pi 5 gemessen, dreimal je Zweig, rund ein Zehntel der Bildrate (49,6/50,0/50,8 gegen 44,0/45,6/45,2 fps unter Volllast) und knapp eine Zehntelsekunde beim Start. Wer ihn abwählt – oder wessen Rechner kein XWayland anbietet –, bekommt das alte Verhalten: dann steht beim Merken *Startzustand gemerkt* – *die Fensterstelle kann diese Anzeige nicht merken*, und Größe, Ansicht und alles Übrige sind festgehalten, nur die Stelle nicht. Die Zeile unter dem Haken sagt, was gerade gilt.

Warum es diesen Umweg überhaupt braucht: es ist keine Lücke in LogRunner, sondern das Wayland-Protokoll. Ein Fenster erfährt dort seine Position nicht und darf sie sich auch nicht setzen – wohin ein Fenster kommt, entscheidet allein der Compositor. Gemessen auf dem Pi mit demselben Testfenster unter beiden Anzeigen: unter X11 meldet es nach einem Verschieben auf 300/200 genau das, unter Wayland weiterhin 0/0.

Wer den Haken abwählt und seine Fenster trotzdem an einen festen Platz bringen will, hat zwei Wege. Beide sind auf dem Raspberry Pi nachgemessen. Mit gesetztem Haken braucht man keinen davon.

Erstens: der Compositor stellt die Fenster. Bei labwc geht das mit einer Fensterregel in

`~/.config/labwc/rc.xml` – LogRunner meldet sich unter dem Namen `LogRunner`, und ein Begleitfenster tut das auch:

```
xml <windowRules> <windowRule identifier="LogRunner" title="LogRunner - Karte"> <action
name="MoveTo" x="1920" y="0" /> </windowRule> </windowRules>
```

In der Fensterleiste steht *LogRunner – Karte*: der Name des Begleitfensters gehört seit b37 zum Titel, und daran lässt sich jedes einzeln ansprechen. Der Titel trägt **keine Versionsnummer** – sonst zeigte die Regel nach der nächsten Beta ins Leere. Das Hauptfenster heißt schlicht *LogRunner*; für eine Regel, die nur dieses meint, gehört `type="normal"` dazu, sonst trifft sie auch das Startfenster.

`ResizeTo` braucht es nicht: die **Größe** merkt sich LogRunner auch unter Wayland selbst, nur die Stelle nicht. Ohne `title` gilt die Regel für alle LogRunner-Fenster. Nach dem Ändern der Datei `labwc --reconfigure` aufrufen – das braucht `LABWC_PID`, sonst bricht es mit *LABWC_PID not set* ab: `LABWC_PID=$(pgrep -x labwc)` `labwc --reconfigure`. Der Platz steht damit fest – auch das Verschieben mit der Maus gilt dann nur bis zum nächsten Start.

Woher die Zahlen kommen: aus LogRunner selbst. Einmal über XWayland starten (unten), die Fenster hinstellen, in jedem *Startzustand merken* – danach steht in jeder `config.toml` unter `[window]` die wirkliche Stelle, und die trägt man in die Regeln ein. Wer das Quellpaket hat, nimmt dafür

`tools/labwc_window_rules.py`: es liest alle Konfigurationen und schreibt den Regelblock in die `rc.xml`.

Zweitens: LogRunner über X11 starten. In einer Wayland-Sitzung läuft es dann über XWayland, die X11-Schicht, die ohnehin mitläuft – und damit kennt das Fenster seine Stelle wieder. *Startzustand merken* hält sie fest und der nächste Start setzt sie, genau wie unter Windows:

```
bash GDK_BACKEND=x11 ~/Programme/LogRunner/LogRunner
```

Dauerhaft wird es in der Verknüpfung, indem die Zeile `Exec=` ein `env` davorbekommt:

```
Exec=env GDK_BACKEND=x11 DISPLAY=:0 /home/USER/Programme/LogRunner/LogRunner
```

DISPLAY=:0 gehört dazu, sonst startet LogRunner nicht. Mit `GDK_BACKEND=x11` sucht GTK einen X11-Bildschirm, und der Desktop gibt einem aus dem Menü gestarteten Programm nicht in jedem Fall ein `DISPLAY` mit. Am Pi nachgestellt: ohne die Angabe endet der Start sofort mit *Gtk-WARNING: cannot open display* und Rückgabewert 1 – kein Fenster, keine Meldung. Welche Nummer die richtige ist, sagt `echo $DISPLAY` in einem Terminal der laufenden Sitzung; in aller Regel ist es `:0`.

Startet trotzdem nichts, fehlt der X11-Schlüssel. XWayland lässt nur Programme herein, die einen Schlüssel aus `~/.Xauthority` vorzeigen können. `xauth list` muss dort eine Zeile für den eigenen Bildschirm zeigen (`rechnernamen/unix:0`). Fehlt sie, kommt beim Start *Authorization required, but no authorization protocol specified*, und auch das sieht von außen wie *tut nichts*. Raspberry Pi OS legt die Datei bei jeder Anmeldung neu an (`/usr/bin/create_xauth`) – ein Ab- und Anmelden ist also die einfachste Reparatur.

Begleitfenster erben den X11-Zweig nicht zuverlässig. Am Pi gemessen: der Hub lief über XWayland und merkte sich seine Stelle, die aus ihm geöffneten Begleitfenster liefen weiter unter Wayland und merkten sich keine. Wer sie mitnehmen will, schreibt denselben `env`-Vorspann in ihre Verknüpfungen – das sind die Dateien `LogRunner-<Name>.desktop` im Programmordner.

Nachgemessen auf dem Pi: Fenster auf 420/260 geschoben, *Startzustand merken*, und genau das stand danach in der `config.toml`. Auf einem Bildschirm ohne Skalierung ist an der Anzeige kein Unterschied zu bemerken; es ist derselbe Weg, den LogRunner auf jedem X11-Desktop geht.

Wie alles hier gilt das je Fenster: jedes Begleitfenster hat seine eigene `config.toml`, und der Hub daneben behält seinen Startzustand, auch wenn der Client nebenan einen neuen bekommt.

Im Dialog *Werkzeuge* ▶ *Anzeige* standen dieselben drei bis 0.9.3b4 als Haken. Sie sind dort entfallen: zwei Wege zu einer Einstellung sind einer zu viel, und der Weg über das Werkzeugmenü liegt hinter dem Menüband, das er ausblenden kann.

Mit *Positionen der Clusterspots nachschlagen* setzt die Karte DX-Meldungen an ihren wirklichen Ort statt in die Mitte des DXCC-Gebiets.

 **Achtung:** Wer nachschlägt, gibt das gerade gearbeitete Rufzeichen an einen fremden Server weiter. Die Passwörter beider Dienste stehen im Klartext in `config.toml` – hier also kein Passwort verwenden, das anderswo noch etwas aufschließt.

xOTA- und SCP-Listen

Werkzeuge ▶ *xOTA/SCP-Listen*

Der Knopf **Update** holt die aktuelle POTA-, SOTA- beziehungsweise IOTA-Liste direkt von der Quelle; danach wird ohne Netz darin gesucht – nach Referenz oder nach Namen, direkt in den Feldern *POTA-Referenz* und *SOTA-Referenz* unter *Allgemeines*. Eine Übernahme setzt zusätzlich den Locator der Referenz, sofern das Locator-Feld noch leer ist. Die Listen liegen dem Programm bewusst nicht bei – sie ändern sich laufend.

Ohne Netz geht es von Hand: die Datei selbst herunterladen und wählen, oder sie unter dem genannten Dateinamen in den Datenordner legen.

Die **Super-Check-Partial-Liste** speist die Rufzeichenvorschläge in der Maske. Sie hält sich mit dem Haken *Selbst aktuell halten* von allein auf Stand: beim Start wird nachgesehen und neu geholt, sobald sie eine Woche alt ist (rund 360 KB je Mal).

LiveLink über mehrere PCs

Werkzeuge ▶ *LiveLink*

Hieß bis Version 0.9.3b8 Netz-Synchronisation. Der Name *LiveLink* ist in allen Sprachen derselbe und wird nicht übersetzt. Eine vorhandene Einrichtung bleibt erhalten: der Abschnitt `[netsync]` einer älteren `config.toml` wird beim ersten Start übernommen.

Ein Rechner im LAN führt als **Hub** den Verteiler, alle anderen hängen als **Clients** daran. Was auf einem PC geschieht, geht an den Hub und von dort an alle übrigen. Der Hub wird per mDNS gefunden; eine von Hand eingetragene Adresse schlägt die Suche.

Der eigentliche Zweck ist die **Maske**: ein Rechner zeigt genau einen Bereich, formatfüllend – die Karte auf dem Bildschirm nebenan, die Band-Map auf dem Rechner links.

Maske	Zeigt
(leer)	Alles – das ganze Fenster, und der Weg zurück
cluster:table	DX-Cluster · Tabelle
cluster:map	DX-Cluster · Band-Map
cluster:stats	DX-Cluster · Bandaktivität
logTable	Log
map	Karte
solar	Bandbedingungen
scope	TRX
workedBands	Gearbeitete Bänder
capture	Gerätebild
qrz	QRZ
special	Ticker

Die Bandzuweisung

Hängen an mehreren Rechnern mehrere Transceiver, muss feststehen, **welcher gemeint ist**. Die Tabelle *Bandzuweisung* legt das je Band fest: ein Auswahlfeld pro Band, zur Wahl stehen die verbundenen PCs.

Daran hängen drei Dinge:

- **Ein angeklickter DX-Spot** stellt das Gerät des zuständigen PCs – auch wenn der Klick in einem Fenster auf einem anderen Rechner passiert ist, und auch dann, wenn im anklickenden Fenster selbst kein Gerät hängt.
- **Die ^-Taste** (links neben der 1) tastet den Sender des PCs, der für das Band in der Eingabemaske zuständig ist – gleich, welches Fenster gerade den Fokus hat. Der Griff zur Maus, um erst das Wasserfall-Fenster nach vorn zu holen, entfällt damit.
- **Frequenz und Betriebsart** eines Client-Geräts laufen in die Eingabemaske des Hubs, wenn dieser Client für das Band zuständig ist. Ohne das bekam ein QSO, das am Client-Gerät gearbeitet und auf dem Hub geloggt wurde, die Frequenz des Hub-Geräts ins Log, sobald dort einmal am Knopf gedreht wurde.

Die Zuweisung führt der Hub und schickt sie an alle; ändern lässt sie sich auf jedem Rechner. Sie steht in `config.toml` unter `[livelink] band_owner`.

Ohne Eintrag folgt ein Spot dem Gerät, das da ist. Klickt jemand eine Cluster-Meldung in einem Fenster ohne eigenes Gerät an, dann dreht das Fenster, an dem eines hängt – gleich ob Hub oder Begleitfenster. Das ist seit 0.9.3b52 so und war vorher eine Lücke: hing der Transceiver an einem Begleitfenster und der Hub hatte keines, drehte gar nichts.

Wer **zwei** Geräte im Netz hat, sollte die Tabelle deshalb ausfüllen. Ohne sie folgen beide, und das ist selten gemeint; mit ihr bekommt genau der zuständige Rechner den Befehl und alle anderen halten sich heraus.

⚠ Sendet ein Gerät auf Zuruf aus dem Netz und reißt die Verbindung, geht es automatisch auf Empfang. Ein „aus“, das nicht mehr ankommt, wäre sonst ein Träger, der stehen bleibt.

Für ein **Begleitfenster auf demselben Rechner** legt der Knopf *Verknüpfung anlegen* eine Startdatei an. Sie startet ein zweites LogRunner mit eigenem Datenverzeichnis unter `~/LogRunner/clients/<Name>/`, das nur den gewählten Bereich zeigt.

Wo die Verknüpfungen liegen: ~/LogRunner/shortcuts/ . Seit 0.9.3b45, und das ist eine Reparatur: vorher lagen sie neben dem Programm. Ein neues Bündel wird daneben ausgepackt und das alte beiseitegeschoben – mit ihm verschwanden die Verknüpfungen aus dem Blick, und die Begleitfenster fehlten nach einem Update im Kasten *Begleitfenster mit starten*.

Beim ersten Start einer neuen Fassung holt LogRunner die Dateien aus dem Programmverzeichnis dorthin und schreibt sie auf das laufende Bündel um. Das zweite ist genauso wichtig wie das erste: in einer Verknüpfung steht der **volle Pfad** des Programms, und eine, die nach einem Update noch auf das alte zeigt, startet still die vorige Fassung. Wer eine Verknüpfung umbenannt hat, findet sie unverändert wieder – sie wird kopiert und nicht neu gebaut, zeigt dann aber weiter auf das alte Programm.

Unter *Verbundene PCs* steht neben jedem Namen, ob es sich um den **Hub** oder einen **Client** handelt. Der Hub kann das nicht selbst herausfinden – sein eigenes Fenster hängt als Client an ihm wie jedes andere –, also sagt es jeder Rechner beim Anmelden selbst. Ein PC mit einem älteren Programmstand steht als *Client* da.

Autostart der Begleitfenster

Führt dieser PC den Hub, kann er die Begleitfenster beim Programmstart gleich mit hochfahren. Der Kasten *Begleitfenster mit starten* zeigt dafür jede Verknüpfung, die im selben Verzeichnis liegt, in das *Verknüpfung anlegen* schreibt.

Presets. Welche Fenster zusammengehören, hängt am Anwendungsfall – ein Contest-Abend will andere Fenster als ein Portabel-Betrieb. Darum gehören die Haken zu einem **Preset**: oben im Kasten wählen, *Neu* (beginnt als Kopie des gewählten), *Umbenennen*, *Löschen*. Angehakt wird, was zu diesem Preset gehört, und das ist schon gespeichert. Unten legt **Autostart ... mit Preset** fest, welches Preset beim Programmstart hochkommt – unabhängig davon, welches gerade oben bearbeitet wird.

Preset starten. Der Knopf rechts in der Preset-Zeile fährt das gewählte Preset sofort hoch – ohne LogRunner neu zu starten. Er räumt dabei auch auf: laufende Begleitfenster, die nicht zu diesem Preset gehören, werden geschlossen. Danach steht auf dem Schirm genau dieses Preset. Vom Contest-Abend zum Portabel-Betrieb ist es damit ein Klick statt dreizehn Fenster von Hand.

Kein Preset gewählt	Dann startet nichts, auch wenn <i>Autostart</i> angehakt ist – und das steht rot neben der Auswahl. Wird das Autostart-Preset gelöscht, bleibt der Autostart ohne Preset; ein anderes wird nicht erraten
Nach dem Update	Eine Autostart-Liste aus einer früheren Version wird zum Preset <i>Standard</i> und ist für den Autostart gewählt – es kommt dasselbe hoch wie vorher
Sofort gespeichert	Beide Häkchen wirken auf der Stelle – <i>Übernehmen</i> braucht es dafür nicht, und ein Haken startet auch keinen Hub neu
Nur auf dem Hub	Der Kasten erscheint nur bei der Rolle <i>Hub</i> . Ein Client, der Clients startete, schickte sie zu einem Verteiler, den es auf diesem Rechner nicht gibt
Einmal je Programmstart	Und erst, wenn der Hub wirklich steht. Ein <i>Übernehmen</i> im Dialog startet nichts nach
Reihenfolge	Die der Liste, also alphabetisch
Fehlende Datei	Bleibt durchgestrichen in der Liste stehen und wird als <i>fehlt</i> gekennzeichnet. Eine umbenannte oder verschobene Verknüpfung soll nicht still verschwinden – sonst käme ein Fenster eines Morgens nicht mehr hoch, und nichts sagte warum
Preset starten	Fährt das oben gewählte Preset hoch – nicht das Autostart-Preset: gestartet wird, was man gerade ansieht. Aktiv, solange der Hub läuft. Unter dem Knopf steht danach, was geschehen ist: wie viele Fenster gestartet wurden, wie viele schon liefen und wie viele geschlossen wurden

Hub beenden	Wird das Fenster geschlossen, das den Hub führt, gehen alle Begleitfenster <i>dieses</i> Rechners mit – ob sie zu einem Preset gehören oder von Hand gestartet wurden. Fenster auf anderen PCs bleiben offen. Das gilt für ein ordentliches Beenden über das Kreuz; stürzt der Hub ab, bleiben die Begleitfenster stehen und verbinden sich wieder, sobald er zurück ist
Was dabei zugeht	Jedes laufende Begleitfenster <i>dieses</i> Rechners, das im Preset nicht vorkommt. Ein Fenster, das in beiden Presets steht, bleibt offen und wird nicht neu gestartet. Fenster auf anderen PCs im Netz bleiben unberührt – genau wie der Transceiver reist auch ein Fenster nicht
Ein Fenster, das nicht zugeht	Wird als solches gemeldet und nicht mitgezählt. Der Hub bittet über die LiveLink-Verbindung; ein Begleitfenster, das gerade nicht verbunden ist, hört die Bitte nicht
Umbenannte Verknüpfung im Preset	Dann wird gar nichts geschlossen, und der Dialog sagt es. Aus einer umbenannten Datei lässt sich nicht ablesen, welches Fenster sie startet – und wer das nicht weiß, darf keines schließen
Starten	Der Knopf rechts in jeder Zeile startet das Fenster sofort – unabhängig vom Autostart-Haken, also auch für ein Fenster, das nicht beim Programmstart mitkommen soll. Aktiv ist er, solange der Hub läuft. Läuft das Fenster schon, wird es nicht ein zweites Mal geöffnet; neben dem Knopf steht dann <i>läuft schon</i> . Eine fehlende Datei hat keinen Knopf

Hinweis: Gesucht wird nach der Endung (`.lnk` unter Windows, `.command` unter macOS, `.desktop` unter Linux) und nicht nach dem Namen – eine umbenannte Verknüpfung ist dieselbe Verknüpfung. In `config.toml` stehen unter `[livelink] autostart_shortcuts` nur Dateinamen; ein Eintrag mit einem Pfad darin wird beim Lesen weggelassen.

⚠ **Achtung:** Die Verbindung ist unverschlüsselt und fragt nach keinem Kennwort. Sie ist fürs eigene LAN gedacht, nicht für ein offenes Netz. Beim ersten Start eines Hubs fragt die Windows-Firewall nach Erlaubnis; unter Linux ist der Port gegebenenfalls von Hand freizugeben (`ufw allow 8765/tcp`), und mDNS braucht dort den `avahi-daemon`.

Gerätebild über eine Capture-Karte

Werkzeuge ▸ Bildaufnahme

Hat der Transceiver einen HDMI-Ausgang und steckt daran eine Capture-Karte, zeigt der Reiter *Gerätebild* dessen Schirm – vollständig, mit Menüs und Filteranzeige. Nötig ist **ffmpeg**; ein Knopf im Panel holt es unter Windows selbst.

Einstellung	Bedeutung
Kamera	Die Capture-Karte. <i>Neu suchen</i> liest die Liste neu ein.
Betriebsart	Übernimmt Format, Bildgröße und Bildrate in die Felder darunter
Durchreichen	Bild unverändert weitergeben: beste Qualität, kaum Rechenzeit, bei 1080p/30 je nach Bildinhalt 10–50 Mbit/s
Neu berechnen	Kleiner fürs WLAN, dafür Rechenzeit. Der große Hebel sind Breite und Bildrate, nicht die Qualitätsstufe.
Bild im Netz anbieten	Aus heißt: der Bildserver hört nur auf <code>127.0.0.1</code>
Zugriff nur mit Schlüssel	Der Schlüssel steht in der Adresse und wird beim Einschalten erzeugt

⚠ **Achtung:** Beim Programmstart *mitlaufen* ist bewusst aus – das Ansprechen mancher Karte weckt den Bildschirm des Transceivers auf.

Ticker – ein Schirm, der nur etwas zeigt

Werkzeuge ▶ Ticker

Drei Dinge, die während des Betriebs oft gebraucht und selten bedient werden:

Anzeige	Was darauf steht
Stationsuhr	UTC links, Lokalzeit rechts, jeweils mit Sekunden; darunter das UTC-Datum mit dem Tag des Jahres und, für die Lokalzeit, das Zonenkürzel mit dem Offset. Daneben Auf- und Untergang von Sonne und Mond für den eigenen Locator, in Ortszeit
Rufzeichen und Locator	Beides vom aktiven Stationsstandort – wer den Standort wechselt, wechselt damit auch, was hier steht
QSOs im Log	Die Gesamtzahl und die der zurückliegenden 60 Minuten
Wetter am Standort	Temperatur, Luftfeuchte, Luftdruck, Wind (Stärke in 10 m Höhe und Richtung) und die Regenwahrscheinlichkeit – jeweils mit dem Verlauf der letzten drei Stunden darunter
Alarm aus dem DX-Cluster	Rufzeichen, Frequenz, Band und Betriebsart einer Meldung, auf die eine Alarmregel angesprochen hat – samt der Regel selbst

Angehakt werden sie einzeln. Sind mehrere gewählt, lösen sie einander **im selben Fenster** ab; wie lange jede steht, sagt das Auswahlfeld daneben – 10 bis 60 Sekunden, für alle gleich. Punkte unter der Kachel zeigen, welche von wie vielen gerade zu sehen ist. Bei einer einzelnen Anzeige wechselt nichts, und das Auswahlfeld ist dann ausgegraut.

Die Uhr führt UTC, und zwar nicht ausgerechnet aus der Lokalzeit: sie liest sie unmittelbar, damit die Anzeige stimmt, egal wie der Rechner eingestellt ist. Sie taktet auf die Sekundengrenze statt im Abstand von einer Sekunde – so läuft sie auch nach Standby und Fensterwechsel nicht nach – und zieht beim Zurückkommen an den Bildschirm sofort nach.

Sonne und Mond stehen neben der Uhr, sobald im aktiven Stationsstandort ein Locator eingetragen ist – gerechnet wird für dessen Mittelpunkt, angezeigt in Ortszeit. Geht die Sonne an einem Tag nicht unter oder gar nicht auf, steht das dort statt einer Uhrzeit; nördlich des Polarkreises ist das die richtige Auskunft. Beim Mond fehlt an rund einem Tag im Monat der Aufgang – er geht täglich später auf, und dann fällt keiner mehr in denselben Tag.

Das Wetter kommt von [Open-Meteo](#) und wird höchstens alle 15 Minuten geholt. **Gefragt wird nur, solange der Haken gesetzt ist** – ohne ihn macht LogRunner dafür keine Verbindung auf. Die Quelle steht unter der Kachel; Open-Meteo gibt seine Daten unter CC BY 4.0 heraus, und die Namensnennung ist die Bedingung dafür. Bleibt die Antwort einmal aus, steht der letzte Stand weiter da und wird nach einer Dreiviertelstunde als alt gekennzeichnet.

Unter jedem Wert seine Linie. Sie zeigt die letzten drei Stunden, und LogRunner führt sie selbst: jeder Abruf wird in einer eigenen kleinen Datei neben dem Logbuch aufgehoben (`data/weather.sqlite`, eine Woche tief), denn Open-Meteo gibt seinen Rückblick nicht so heraus, dass eine Anzeige ihn nebenbei mitholen könnte. Beim Luftdruck ist die Linie der eigentliche Nutzen der Zahl: 1008 hPa sagt für sich genommen nichts, 1008 hPa nach 1016 sagt, dass eine Front unterwegs ist.

Nach einem frischen Start ist die Linie noch kurz – neben der Quelle steht dann, wie weit sie wirklich zurückreicht, damit aus zwei Punkten niemand einen Temperatursturz liest. Die Windrichtung bekommt als einzige keine: sie läuft im Kreis, und eine Linie von 350° auf 10° zeichnete einen Dreh um das ganze Zifferblatt, den es nicht gab. Die Datei zu löschen kostet die Linien und sonst nichts.

Der Alarm hat Vorrang. Er läuft nicht im Wechsel mit, sondern unterbricht: Sobald eine Meldung eintrifft, auf die eine Ihrer Alarmregeln aus [Werkzeuge ▶ DX-Cluster](#) anspricht, steht sie im Ticker – mit rotem Rand, für die eingestellte Anzeigedauer. Danach geht der Wechsel weiter, wo er war. Ein neuer Alarm setzt die Zeit zurück, sodass immer der letzte zu sehen ist.

Dafür holt der Ticker die Meldungen selbst, solange der Haken steht – auch auf einem Fenster, das den DX-Cluster gar nicht zeigt. Ohne den Haken unterbricht nichts, und ohne eingeschalteten Cluster gibt es nichts zu

unterbrechen.

Auf einem Client funktioniert das genauso. Er hat keine eigene Verbindung zu einem Knoten – die Meldungen bekommt er vom Hub, und **dessen Alarmregeln gelten**. Im Alarmbereich des Cluster-Panels stehen sie zum Nachlesen: worauf etwas anspringt, wird am Hauptrechner eingerichtet und nicht an jedem Schirm einzeln.

Zu hören ist der Alarm nur auf dem Hub. Ein Client zeigt ihn – als Zahl am Reiter und als Kachel im Ticker –, aber er klingelt nicht. Ein Operator, mehrere Schirme: ein Ton, der aus zwei Zimmern kommt, sagt nicht mehr als einer.

Der Filter bleibt dagegen jedem Schirm sein eigener. Der Hub gibt vor, was es gibt; was ein Schirm davon zeigt, entscheidet er selbst – der Wandschirm im Nebenzimmer darf auf 20 m stehen, während am Gerät alles läuft.

Deshalb sind die Felder in *Werkzeuge* ▶ *DX-Cluster* auf einem Client ausgegraut: Knoten, Alarmregeln und Ton kommen von dort, und ein Feld, das sich bedienen lässt und nichts bewirkt, ist die Sorte Anzeige, wegen der man den Fehler beim Gerät sucht. **Aufmachen lässt sich der Dialog aber** – nachsehen ist etwas anderes als verstellen, und wer wissen will, an welchen Knoten der Hub hängt oder welche Regeln gelten, findet es dort.

Wenn ein Client seinen eigenen Cluster fahren soll, steht ganz oben im Dialog der Haken **Hub folgen**. Er ist die eine Ausnahme von der Sperre und ab Werk gesetzt. Nehmen Sie ihn ab, arbeitet dieses Fenster **im DX-Cluster** eigenständig: eigene Knoten, eigener Filter, eigene Alarmregeln, eigener Ton – der Dialog geht dafür wieder auf. Alles Übrige spiegelt es weiterhin, Logbuch, Karte und Eingabemaske eingeschlossen; der Haken betrifft nur den DX-Cluster.

Beim Umlegen wird die Meldungsliste geleert – sie hat ab dann eine andere Quelle, und zwei Zeiten in einer Tabelle sieht man ihr nicht an. Und ohne eingeschalteten eigenen Cluster verschwindet der Reiter *DX-Cluster*: dann gibt es keine Meldungen mehr, weder eigene noch die des Hubs.

Der eigene Schirm. Der Zweck der Sache ist ein zweites Fenster, das nichts anderes zeigt: ein Monitor über dem Gerät, auf dem die Uhr steht. Dafür gibt es im LiveLink-Dialog unter *Verknüpfung für ein Begleitfenster* die Maske *Ticker* – oder auf der Kommandozeile `--client Uhr --mask special`. Auch ein **Client** kann sie zeigen: Rufzeichen und QSO-Zahlen holt er sich dann vom Hub, denn ein Client führt kein eigenes Logbuch.

Jedes Fenster hat seine eigene Auswahl. Ein Begleitfenster hat sein eigenes Datenverzeichnis – der Schirm an der Wand darf also die Uhr zeigen, während das Hauptfenster den QSO-Zähler stehen hat.

4.3 Workflow 3 – Contestbetrieb

Einen Contest beginnen. Im Reiter *Contest* der Eingabemaske (oder unter *Werkzeuge* ▶ *Conteste*) einen neuen Contest anlegen:

Feld	Bedeutung
Contest	Das ADIF-Kürzel, etwa CQ-WW-SSB
Austausch	Der feste Teil des eigenen Austauschs (Zone, Bezirk, Name ...)
Zählt eine laufende Nummer	Aus für Conteste, die keine kennen (CQ WW tauscht Rapport und Zone)
Nummer je Band	Getrennte Zählung je Band
Dupe zählt	<i>je Band und Betriebsart (Voreinstellung), je Band oder einmal überhaupt</i>

Der Betrieb. *Betrieb* ▶ ▶ *Contest* schaltet die Maske auf die **Contest-Zeile** um. Sie zeigt nur, was auf der Luft durchgegeben wird, in genau dieser Reihenfolge: Rufzeichen · RST ges. · Nr. ges. · RST empf. · Nr. empf. · Austausch empf. Der Tabulator läuft damit mit dem Ohr mit, Enter loggt aus jedem Feld, und der Cursor springt danach von selbst zurück auf das Rufzeichen. **Esc** verwirft die Eingabe und tut dasselbe.

Die eigene laufende Nummer wird gezählt, nicht getippt: sie ist $\max(stx)+1$ über die Sitzung und wird beim Loggen vergeben. Von Hand einzutragen ist sie im Reiter *Contest*.

Dupes. Steht die Station in diesem Contest schon im Log, erscheint eine rote Plakette „*Schon gearbeitet – hh:mm UTC auf xx*“, und das QSO wird **nicht** geloggt. **Strg + Alt + Enter** loggt es trotzdem – dieselbe Tastenkombination, die N1MM Logger+ dafür benutzt.

⚠ **Achtung:** Ob eine Doppelverbindung zählen darf, ist die Regel des Contests, nicht die dieses Programms. Deshalb gibt es den Weg daran vorbei – er ist nur nicht der, der aus Versehen passiert.

Der Knopf *Volle Maske* holt Name, QTH, Locator und Kommentar zurück, ohne den Contest zu beenden – für das eine QSO zwischendurch, das keines ist.

Nach dem Contest. *Werkzeuge* ▶ *Conteste* listet alle gelaufenen Conteste. Von dort aus:

- **Wieder aufnehmen** – der zweite Morgen eines Wochenendes; Nummern und Dupes zählen weiter, wo sie aufgehört haben.
- **Löschen** – entfernt nur die Sitzung. Die QSOs bleiben im Log.
- **Cabrillo-Einreichung** – Kopfdaten ausfüllen (Rufzeichen, Kategorien, gemeldete Punktzahl, Operatoren, Club, Anschrift, Anmerkungen), *Vorschau* prüfen, *Datei schreiben*.

⚠ **Achtung:** Der Cabrillo-Kopf ist Ihre Erklärung – nichts davon lässt sich aus dem Log erschließen, und ein Einreichungsroboter entscheidet daran. Welche Form des Austauschs ein bestimmter Contest erwartet, ist dessen eigene Regel: die QSO-Zeilen vor dem Absenden gegen die Ausschreibung halten.



Abb. 10: Die Contest-Zeile mit Rufzeichen, beiden Rapporten und den Austauschfeldern

4.4 Workflow 4 – Nachsehen, was schon gearbeitet ist

Gearbeitete Bänder. Der gleichnamige Reiter zeigt zum Rufzeichen in der Maske eine Bandmatrix mit den Zuständen *Gearbeitet*, *Bestätigt* und *Noch nicht*. Der Haken *Nach Mode* schlüsselt zusätzlich nach CW, Phone und Digital auf. Ein Klick auf eine Bandzelle öffnet die QSO-Liste dazu; von dort lässt sich ein QSO direkt zum Bearbeiten öffnen.

Log-Tabelle. Über der Tabelle steht die Filterzeile: Rufzeichen, Band, Mode, Submode, Contest, Von und Bis. Jede Spalte ist durch Klick auf ihre Überschrift sortierbar, ein zweiter Klick kehrt die Richtung um. Der Knopf *Spalten* wählt aus, was die Tabelle zeigt, und ordnet die Spalten; mindestens eine muss stehen bleiben.

Zahlen stehen in der eingestellten Sprache. Frequenz, Entfernung, Sendeleistung und die Antennenwinkel folgen der Sprachwahl in der Kopfzeile – auf Deutsch also 14,074 (MHz) und 14.074.000 (Hz), auf Englisch umgekehrt. DXCC-Nummer, Zonen und die laufenden Nummern eines Contests bleiben ohne Trennzeichen: das sind Kennungen, keine Größen. **Das Eingabefeld der Maske macht das nicht mit** – dort steht die Frequenz immer mit Punkt, weil ein Zahlenfeld des Browsers nichts anderes annimmt.

Der Rollbalken deckt das ganze Logbuch ab – seit 0.9.3b38, davor waren es die letzten 500 QSOs, was bei einem großen Log niemand sah außer beim Versuch, ans Ende zu scrollen. Die Zeilen werden nachgeladen, während man rollt; wo noch keine da ist, steht so lange ein grauer Balken statt einer falschen Angabe. Sortiert und gefiltert wird dabei über den **gesamten** Bestand, nicht über das, was gerade zu sehen ist. Bei sehr großen Logbüchern (jenseits

von hunderttausend QSOs) kann ein Sprung mitten in eine Tabelle, die nach einer selten benutzten Spalte sortiert ist, eine knappe Sekunde brauchen – die Vorgabe *UTC absteigend* und die Filterzeile antworten sofort.

Groß- und Kleinschreibung spielt beim Rufzeichen nirgends eine Rolle – seit 0.9.3b38 auch beim Nachschlagen, nicht mehr nur in der Filterzeile. Ein `db0jr`, das aus einem alten Bestand klein im Logbuch steht, wird damit von *worked before* und der Band-Matrix gefunden wie `DB0JR`.

Statistik. Werkzeuge ▶ *Statistik* – QSOs gesamt, Rufzeichen, DXCC-Entitäten, Kontinente und bestätigte Verbindungen als Kacheln, dazu die Aktivität über 24 Stunden / 7 Tage / 30 Tage / 12 Monate, eine Band-gegen-Mode-Matrix, Auswertungen nach Kontinent, Jahr, Standort und Operator sowie der IOTA-Fortschritt.

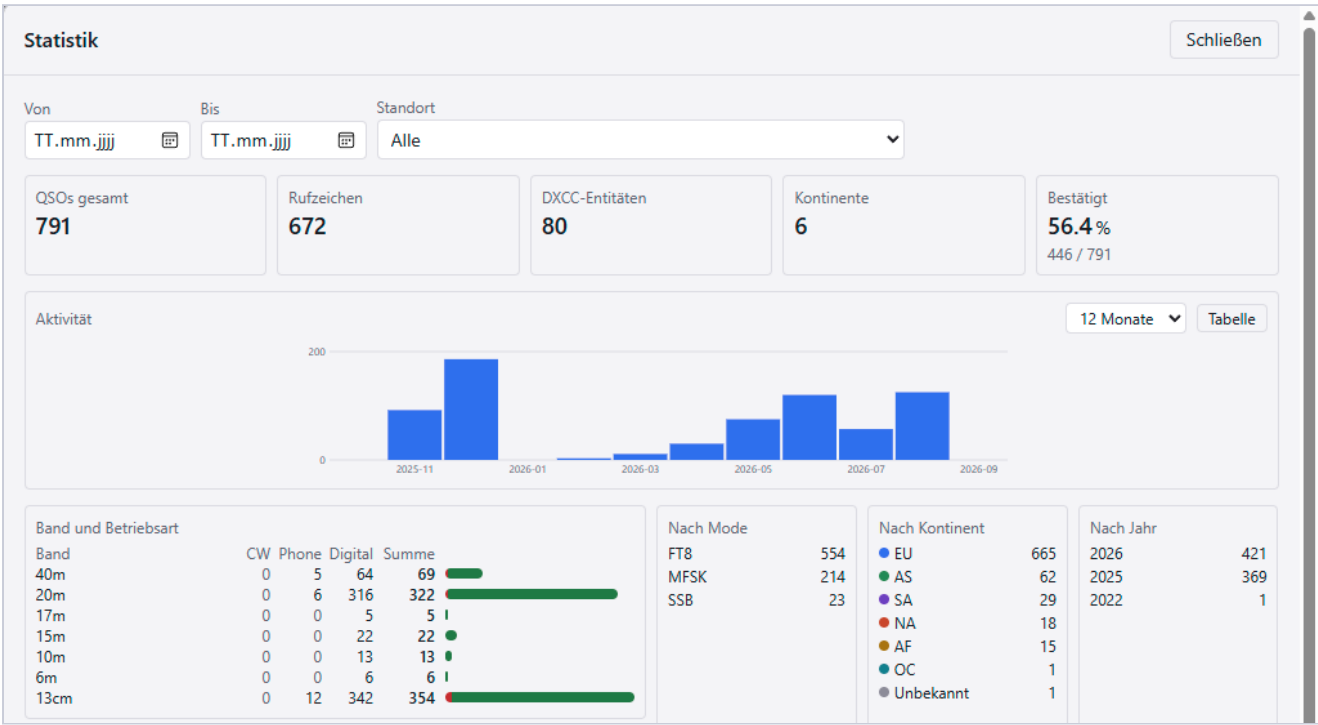


Abb. 11: Die Statistik mit Kacheln, Zeitverlauf und Band-Mode-Matrix

Ein QRZ.com-Fenster für alle Fenster auf einem Rechner. Wer aus einem Begleitfenster heraus ein Profil öffnet, bekommt es im QRZ-Fenster des Hauptfensters gezeigt – es geht kein zweites auf. Läuft ein Client auf einem **anderen PC**, öffnet der weiterhin sein eigenes; das Fenster des Hubs stünde dort auf einem Schirm, auf den niemand sieht.

4.5 Workflow 5 – Die Karte

Der Reiter *Karte* zeigt, wo die Gegenstation sitzt – grob nach Landeskenner, genau nach Locator –, die eigene Station und die Großkreislinie dazwischen. Küsten, Inseln, Eis, Seen, Riffe, Flüsse und Landesgrenzen liegen als Vektordaten bei; es wird kein Kartendienst befragt.

Drei Ansichten:

Ansicht	Besonderheit
Karte	Flache Weltkarte, ziehen zum Verschieben, Mausrad zum Zoomen
Globus	Die Großkreislinie ist der Bogen, der sie wirklich ist; ziehen zum Drehen
Azimut	Die ganze Erde um die eigene Station: jede Gerade durch die Mitte ist eine Antennenrichtung, jeder Ring eine Entfernung, der Rand der Gegenpunkt in 20.015 km

Einblendbare Ebenen: **Graulinie** (Dämmerungszone und Nachtseite zur laufenden UTC-Zeit), **Präfixe** (DXCC-Gebiete, anklickbar), **CQ-Zonen** und **ITU-Zonen** (mit Grenzen und Nummern, anklickbar), **Locator-Raster** (Feinheit folgt dem Zoom), **Orte** (Städte und Gemeinden nach Zoomstufe), **IOTA** (gefüllter Ring = gearbeitet, hohler = noch nicht), **MUF** (Höhenlinien der höchsten nutzbaren Frequenz – siehe unten), **DX-Spots** und **Log-Einträge** (die QSOs, die die Tabelle gerade zeigt, mit ihrem Filter).

Ein Klick auf einen Spot oder einen Log-Eintrag übernimmt ihn in die Eingabemaske.

Achtung: Ohne eigenen Locator im Stationsstandort gibt es keine Entfernung, keine Peilung, keine Linie – und die Azimutkarte hat keinen Mittelpunkt.

Die MUF-Ebene. Der Knopf *MUF* legt Höhenlinien über die Karte: die höchste Frequenz, die die Ionosphäre an dieser Stelle gerade noch zurückwirft. Praktisch gelesen sagt die **14,0-Linie**, bis wohin 20 m gerade trägt – wer innerhalb von ihr sitzt, hat das Band. Sechs Stufen von 10,1 bis 28 MHz, kalte Farben für die niedrigen, warme für die hohen Bänder.

Zwei Dinge gehören dazu, sonst wird die Karte falsch gelesen:

- **Die Zahl gilt für einen Sprung von 3000 km**, nicht für Ihre Strecke. Sie sagt, *wo* die Ionosphäre wie viel trägt, nicht *ob* eine bestimmte Verbindung zustande kommt. Über zwei Sprünge oder über 500 km kommt etwas anderes heraus.
- **Alter zählt.** Die Daten kommen von prop.kc2g.com, das alle 15 Minuten neu rechnet; an der Legende steht, wie alt der Stand ist. Ab einer Stunde werden die Linien **gestrichelt und blasser** gezeichnet und das Alter rot – eine drei Stunden alte MUF-Kontur ist nicht ungefähr richtig, sie ist falsch.

Ohne Verbindung bleibt der zuletzt geholte Stand stehen, mit seinem Alter daneben; *Aktualisieren* an der Legende fragt von Hand nach.

Hinweis: Die MUF-Ebene greift ins Netz und ist deshalb ab Werk gesperrt. Beim ersten Einschalten steht an der Legende *Abruf nicht freigegeben* und daneben der Knopf **Daten abrufen** – ein Klick gibt die Quelle frei und holt sofort. Danach schaltet der Kartenknopf nur noch die Anzeige. (In der `config.toml` tut `enabled = true` im Abschnitt `[muf]` dasselbe.)

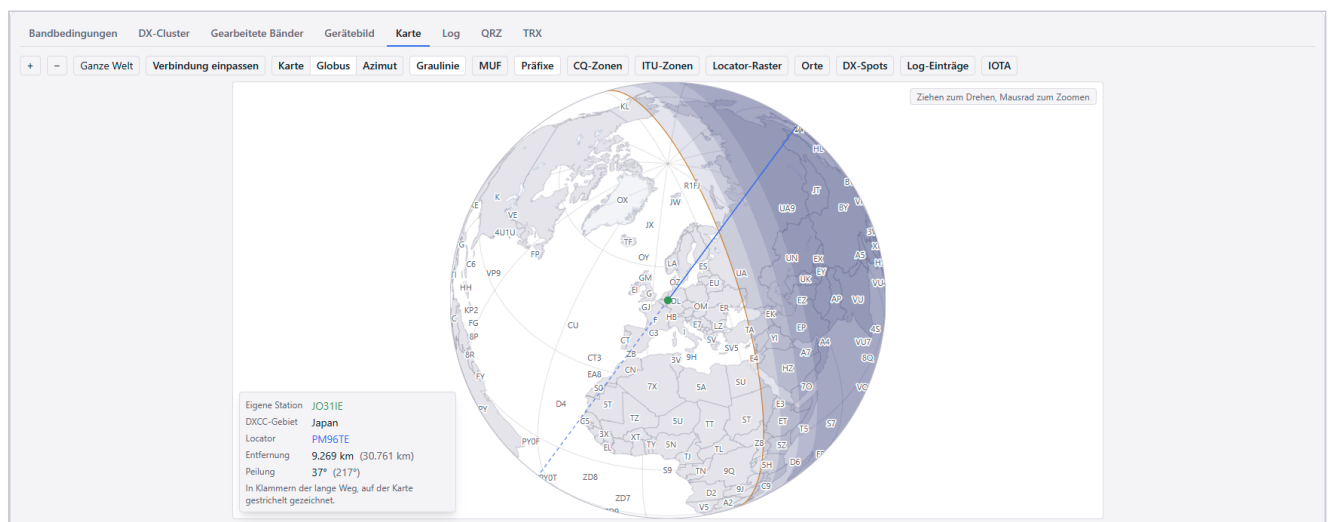


Abb. 12: Die Weltkarte als Globus mit Großkreislinie und eingeblendeten Präfixen

4.6 Workflow 6 – Bandbedingungen

Der Reiter *Bandbedingungen* beantwortet drei verschiedene Fragen, und sie stehen in dieser Reihenfolge untereinander:

Bereich	Die Frage dahinter
Die Zahlen	Wie ist die Lage überhaupt?
Pfadindikator	Welche Bänder tragen zu dieser Station?
Bandindikator	Wohin komme ich gerade überhaupt?

Der erste kommt von N0NBH und ist eine Messung, die beiden anderen sind eine Rechnung gegen die MUF-Konturen von KC2G. Das ist kein Detail, sondern der Unterschied zwischen *wie steht die Sonne* und *kommt mein Signal dort an*.

Die Zahlen

Auf Wunsch geholt von hamqsl.com: Sonnenfluss, Sonnenflecken, A- und K-Index, MUF, Röntgenfluss, Zustand des Erdmagnetfelds und Störpegel, dazu die Einschätzung je Bandgruppe für Tag und Nacht. Abgefragt wird höchstens stündlich; die Antwort wird lokal abgelegt und steht auch ohne Verbindung – dann mit ihrem Zeitstempel und dem Vermerk *veraltet*.

LogRunner legt zusätzlich einen Messwert je Stunde ab und zeigt daraus **Trendpfeile** und einen Gesamttrend (0–100), gedacht zum Vergleich mit sich selbst.

Die MUF-Karte kommt eventuell woanders her. hamqsl.com liefert dieses eine Feld seit geraumer Zeit leer. Bleibt es leer und ist die MUF-Ebene der Karte freigegeben (Abschnitt 4.5), holt LogRunner den Wert bei prop.kc2g.com nach – von der Ionosonde, die Ihrem Locator am nächsten steht. Die Karte trägt dann • **KC2G** und nennt im Tooltip Station, Entfernung und Alter; unter dem Panel steht eine zweite Quellenzeile. Ohne eigenen Locator, ohne Sonde im Umkreis von 3000 km oder solange die Quelle nicht freigegeben ist, bleibt der Strich stehen – jeweils mit einem Hinweis, welcher der drei Gründe es ist. Eine Zahl von weit her wäre keine Antwort auf „was geht hier gerade“, und ein Strich ohne Grund sieht aus wie ein Fehler.

Pfadindikator

Dieselbe Frage ein zweites Mal, aber für **eine** Strecke. Wofür gerechnet wird, sagen die beiden Felder daneben:

Trägt jetzt [Rufzeichen od. Präfix] / [QTH-Locator]

Ins **erste** kommt ein Rufzeichen oder auch nur ein Präfix – VK genügt, VK3ABC geht ebenso. Bleibt es leer, gilt das Rufzeichen aus der Eingabemaske; das ist der Normalfall beim Loggen, und das Feld ist für die Frage daneben da: *und was wäre, wenn ich jetzt VK riefe?*

Eine Eingabe im Rufzeichenfeld der QSO-Maske stellt diese Nebenfrage zurück und führt den Pfadindikator mit der neuen Station nach: was zu ihr gerade trägt, ist beim Loggen die häufigste Frage, und sie soll ohne einen zweiten Handgriff beantwortet sein.

Ins **zweite** kommt der QTH-Locator der Gegenstation, und er ist die genauere Angabe: ein Rufzeichen nennt bestenfalls ein Land, und die Mitte von W oder UA liegt tausend Kilometer von der Station entfernt. Steht ein Locator da, wird gegen den Ort gerechnet statt gegen das Land – er schlägt das Rufzeichen.

Ist der Locator durch das Rufzeichen der Eingabemaske schon bekannt, steht er grau als Platzhalter im Feld. Das heißt *dieser gilt gerade*, und man muss ihn nicht erst wegräumen, um einen anderen einzutragen – dieselbe Regel wie beim Rufzeichenfeld daneben.

Und tippt man im Pfadindikator ein eigenes Rufzeichen ein, sucht LogRunner dessen Locator selbst heraus. Vier Quellen, in dieser Reihenfolge:

Reihenfolge	Quelle
zuerst	der Abfrage-Zwischenspeicher : wurde das Rufzeichen schon einmal bei QRZ.com oder HamQTH nachgeschlagen, steht sein Locator dort
dann	das eigene Log : das jüngste QSO mit dieser Station, das einen Locator trägt
dann	QRZ.com oder HamQTH , frisch abgefragt – sofern eingerichtet und das Netz erreichbar ist
zuletzt	cty.dat : die Mitte des DXCC-Gebiets, in einen Locator umgerechnet

Die ersten beiden und die letzte antworten von der Platte und sofort. Nur die dritte kann Sekunden brauchen, und sie wird deshalb **nur hier** gestellt – nicht bei jedem Anschlag in der Eingabemaske, und auch nur für ein **ganzes** Rufzeichen: ein Präfix hat bei keinem der Dienste einen Eintrag.

Die letzte Quelle sieht anders aus als die anderen drei. Ein Locator aus dem Zwischenspeicher, dem Log oder von QRZ beschreibt eine Station und steht deshalb als Wert im Feld. Die Mitte eines Landes ist keine Station – sie steht nur **grau als Platzhalter** darin, und der Text unter dem Mauszeiger sagt, woher sie kommt. Gerechnet und gezeichnet wird trotzdem gegen sie; das Fadenkreuz ✚ in der Zeile daneben sagt weiterhin *Landesmitte, nicht Station*.

Dieselbe Anzeige gilt für das Rufzeichen aus der **Eingabemaske**: steht dort eine Station, deren Locator LogRunner kennt, erscheint er hier grau als Platzhalter – auch dann, wenn er in der Maske selbst gar nicht steht, weil er aus dem Log oder dem Zwischenspeicher kommt. Gezeigt wird immer der Locator, mit dem tatsächlich gerechnet wurde, und der Text unter dem Mauszeiger nennt seine Quelle.

Das gilt auch, wenn der Pfadindikator gerade **nichts beurteilen kann** – weil die MUF-Konturen fehlen oder der Abruf nicht freigegeben ist. Welcher Ort gemeint ist, hängt nicht an der Ionosphäre; ein leeres Feld hieße dort *kein Locator bekannt*, und das wäre etwas anderes.

Findet sich kein Stationslocator, wird das Feld geleert – auch dann, wenn eben noch einer darinstand. Das ist keine Nachlässigkeit, sondern der Punkt: der Locator schlägt das Rufzeichen, und ein alter Locator neben einem neuen Rufzeichen würde die Rechnung gewinnen. Angezeigt stünde VK, gerechnet würde nach JO31.

Wer den Locator kennt, trägt ihn danach von Hand ein – das überschreibt LogRunner nicht.

Zwei Feinheiten zum Zwischenspeicher und zum Log: der Zwischenspeicher sucht über die **Grundform** des Rufzeichens (DL/W1AW/P ist dort W1AW, denn die Lizenz ist dieselbe), das Log über **genau das getippte** (DL/W1AW/P stand woanders als W1AW, und der Locator von damals gehört dorthin). Für ein Rufzeichen mit angehängtem Präfix oder Suffix wird der Zwischenspeicher allerdings **gar nicht** nach dem Ort gefragt: der Datensatz von W1AW nennt Connecticut, und genau dort ist DL/W1AW/P nicht. Dann gilt das Land, das der angehängte Präfix nennt.

Solange das, was im Locatorfeld steht, noch kein vollständiger Locator ist – JO3 etwa –, bekommt das Feld einen roten Rand, und es gilt weiter, was ohne ihn galt. Zwei bis acht Zeichen sind einer.

Das × räumt beide Felder wieder weg – sie stellen **eine** Frage, und wer sie zurückstellt, stellt sie ganz zurück.

Zwei Linien auf der Karte, und sie gehen sich nichts an

Der QSO-Pfad ist die Linie zu der Station, die gerade in der QSO-Eingabemaske steht: durchgezogen der kurze Weg, gestrichelt der lange. Er erscheint, sobald dort ein Rufzeichen oder ein Teil davon steht, und er bleibt stehen, solange etwas darinsteht – gleich, wie es dorthin kam: getippt, durch Anklicken einer DX-Meldung in der Tabelle oder auf der Karte, oder durch Klick auf einen Logbucheintrag.

Die Linie des Pfadindikators ist die Strecke, nach der im Pfadindikator gefragt wurde – gepunktet und in einer eigenen Farbe. Sie erscheint, sobald in einem der beiden Felder etwas steht oder auf die Karte geklickt wurde, und **nur bei eingeschalteter MUF-Anzeige**: sie ist die Strecke, über die der Indikator seine Bänder beurteilt, und ohne Konturen gibt es nichts zu beurteilen.

Keine der beiden nimmt der anderen etwas weg. Wer mitten in einem QSO wissen will, was gerade nach Japan trägt, tippt es in den Pfadindikator oder klickt auf die Karte – der QSO-Pfad bleibt, wo er ist. Bis zum 02.09.2026 war das anders: es gab nur eine Linie, und LogRunner fragte erst zurück, ob die Eingabemaske noch gebraucht wird. **Diese Rückfrage gibt es nicht mehr.**

Unter der Karte stehen beide Strecken untereinander: oben das QSO mit Land, Locator, Entfernung und Richtung, darunter abgesetzt die Strecke des Pfadindikators.

Ist die MUF-Anzeige eingeschaltet, sitzen auf der Linie des Pfadindikators die Reflexionspunkte an der F2-Schicht – je Sprung einer, in derselben Farbe wie die Linie. Der begrenzende, an dem die Strecke zumacht, bekommt einen bernsteinfarbenen Ring. Sie liegen dort, wo der Indikator rechnet: hat er keine eigene Frage, ist das der QSO-Pfad, sonst seine eigene Linie.

Der Klick auf die Karte

Der bequemste Weg zu einem Locator ist ein Klick auf die Karte. Solange die MUF-Anzeige an ist, setzt ein Klick irgendwohin den Locator dieser Stelle in das Feld des Pfadindikators und zieht dessen Linie dorthin. Ein Rufzeichen, das im Pfadindikator stand, wird dabei gelöscht – es hat mit dem geklickten Punkt nichts zu tun. Der QSO-Pfad bleibt unberührt.

So genau, wie die Ansicht es hergibt: auf der ganzen Weltkarte zwei Zeichen, hineingezoomt vier, ganz nah sechs – dieselbe Stufe, in der die Karte auch ihr Locorraster zeichnet. Sechs Zeichen aus einem Klick auf die Weltkarte wären zwei geratene.

Ein Zug über die Karte ist kein Klick, und eine angeklickte DX-Meldung ist kein Kartenpunkt: die meint man selbst, und sie geht wie immer in die QSO-Eingabemaske.

Drei Wege führen von einem Klickpunkt wieder weg: das × im Pfadindikator, ein eigener Eintrag in eines seiner beiden Felder – und **eine Eingabe im Rufzeichenfeld der QSO-Eingabemaske.** Wer dort ein Rufzeichen einträgt, will wissen, was zu *dieser* Station gerade trägt; der Pfadindikator wird dann mit ihr nachgeführt, und seine beiden Felder stehen wieder leer. Der QSO-Pfad auf der Karte ändert sich dabei nicht – er folgt der Eingabemaske ohnehin.

Das bloße Leerräumen der Maske – nach *Eingabe leeren* oder nach dem Loggen – zählt nicht: das ist kein neues Ziel, sondern gar keines, und eine offene Frage im Pfadindikator bleibt dann stehen.

Ein Kästchen je Band von 160 m bis 6 m: ein gefüllter Punkt ● heißt offen, ein halber ◐ grenzwertig, ein leerer ○ zu. 8 m fehlt mit Absicht – das Band ist in keiner der drei ITU-Regionen dem Amateurfunk zugewiesen und steht im Logbuch nur, weil ADIF es für die wenigen nationalen Versuchsgenehmigungen führt. In der Zeile darüber stehen die Zahl der Sprünge, die Entfernung, die Strecken-MUF und das Alter der Daten; der Mauszeiger auf einem Kästchen nennt den Abstand zur MUF und bei einem geschlossenen Band den Ort auf der Strecke, an dem sie zumacht.

Der Unterschied zur Tabelle darüber ist wesentlich: die stammt von NONBH und sagt, wie die Bedingungen allgemein sind. Dieser Block rechnet gegen die MUF-Konturen von KC2G entlang des Großkreises zur Gegenstation und sagt, ob die Ionosphäre **dort** die Frequenz zurückwirft. Eine Verbindung sagt er damit nicht voraus: Dämpfung, Rauschen, Antennen und Leistung sind nicht enthalten, und ein Band kann offen und trotzdem unbrauchbar sein. Wer eine echte Streckenbilanz braucht, findet sie im HF Propagation Planner auf prop.kc2g.com.

Zwei Einschränkungen nennt die Anzeige selbst. Liegt mehr als die Hälfte der Strecke im Tageslicht, werden 160 m, 80 m und 40 m gedämpft dargestellt statt grün – die D-Schicht frisst sie am Tage auf, auch wenn die MUF sie trüge. Und oberhalb von 10 m rechnet das Verfahren nur die F2-Schicht: auf 6 m heißt **zu** also **kein F2**, nicht *keine Öffnung*, denn Sporadic-E und Aurora kennt es nicht. Ein *offen* dort ist umgekehrt eine ernste Auskunft.

Bandindikator

Darunter steht dieselbe Ampel noch einmal als Raster: senkrecht die Kontinente – alphabetisch AF, AN, AS, EU, NA, OC, SA –, waagrecht die Bänder, in denselben Kästchen wie im Pfadindikator darüber. Diese Tabelle hängt an

keinem Rufzeichen; sie beantwortet die andere Frage, nämlich *wo komme ich überhaupt gerade hin*. Eine leere Zelle heißt nicht *zu*, sondern *keine Auskunft*.

Ein Kontinent ist kein Ort, und die Anzeige behauptet das auch nicht. Gerechnet wird gegen mehrere Punkte je Kontinent – für Asien etwa Japan, Ostchina, Indien, Sibirien und den Nahen Osten –, und gezeigt wird je Band der **beste** davon. Ein ● heißt also: *irgendwo dort trägt es*, nicht *überall*. Welcher Punkt es trägt, sagt der Mauszeiger, zusammen mit Entfernung, Richtung und Strecken-MUF – damit lässt sich die Antenne drehen. Der Umweg über mehrere Punkte ist nötig, weil ein Kontinent über viele Zeitzonen reicht: zwischen Japan und der Türkei liegt zur selben Stunde die halbe Ionosphäre, und ein Punkt in der Mitte beantwortet keine der beiden Fragen.

Gerechnet wird gegen dieselben Konturen, die auch die MUF-Ebene der Karte zeichnet, und deshalb gilt hier alles, was dort steht: der Abruf ist ab Werk gesperrt, und ohne freigegebene Quelle bleibt der Block leer und sagt das auch.

Die Punkte auf der Streckenlinie

Ist die MUF-Ebene eingeschaltet, markiert die Karte auf der Linie zur Gegenstation kleine Punkte – einen je Sprung.

Sie liegen dort, wo der Strahl an der F2-Schicht **gespiegelt** wird, also in der *Mitte* jedes Sprungs – nicht dort, wo er den Boden wieder berührt. Das ist die Stelle, an der der Zustand der Ionosphäre über diesen Sprung entscheidet, und genau dort rechnet der Pfadindikator.

Daraus folgt, wie sie zu lesen sind:

- Die **Zahl** der Punkte ist die Zahl der Sprünge.
- Der **Abstand zwischen zwei benachbarten** Punkten ist genau eine Sprungweite. Daran lässt sie sich ablesen.
- Der Abstand **von der eigenen Station zum ersten** Punkt ist dagegen nur eine **halbe** Sprungweite – und am anderen Ende ebenso. Wer von dort misst, kommt auf die Hälfte.

Ein Beispiel: 9106 km nach Thailand sind drei Sprünge zu je 3036 km. Die drei Punkte liegen bei 1518 km, 4553 km und 7589 km – zwischen ihnen jeweils 3036 km, zu den Enden hin je 1518 km.

Einer der Punkte kann **bernsteinfarben und größer** sein: das ist die engste Stelle des Wegs, der Sprung mit der niedrigsten MUF. Er bestimmt die ganze Strecke, denn ein Sprung, der nicht zurückkommt, macht den Weg zu – die Antwort auf „warum ist 20 m zu“ ist damit ein *Ort*. Liegt der ganze Weg in derselben Konturstufe, gibt es keine engste Stelle; dann bleiben alle Punkte blau, und das ist keine Fehlanzeige, sondern die richtige Auskunft.

Ein Fenster für einen einzelnen Teil

Auf einem Begleitfenster mit `--mask solar` lässt sich im Rechtsklick-Menü wählen, welcher der drei Teile allein zu sehen ist – *Sonnenaktivität*, *Pfadindikator*, *Bandindikator* oder *Detail* für alles zusammen. Das Fenster misst danach seine Höhe nach. Abschnitt 3.2 nennt die Einträge im Zusammenhang, Abschnitt 4.2 die LiveLink dahinter.

Woher die Zahlen kommen

Der technische Unterbau, für den Fall, dass eine Anzeige einmal nicht das sagt, was sie sagen sollte.

Anzeige	Quelle	Abruf	Liegt in
Zahlen, Bandtabelle, VHF	hamqsl.com (NONBH), <code>solarxml.php</code>	höchstens stündlich	<code>data/solar.json</code>
Trendpfeile, Gesamttrend	dieselben Zahlen, selbst mitgeschrieben	ein Datensatz je Stunde, 30 Tage	<code>data/propagation.sqlite</code>
MUF-Karte (Ersatzquelle)	prop.kc2g.com, <code>api/stations.json</code>	mit den Konturen	<code>data/muf.json</code>

Anzeige	Quelle	Abruf	Liegt in
Pfad- und Bandindikator, MUF-Ebene der Karte	prop.kc2g.com, <code>mufd-normal-now.geojson</code> und <code>fof2-normal-now.geojson</code>	alle 15 Minuten	<code>data/muf.json</code>

Beide Abrufe sind ab Werk **gesperrt** und werden einzeln freigegeben: `[solar] enabled` für die Zahlen, `[muf] enabled` für die Konturen. Gesperrt heißt: nichts wird geholt, ein vorhandener Zwischenspeicher aber weiter angezeigt. Was einmal geholt wurde, gehört Ihnen.

Wie aus einer Kontur eine Ampel wird, in fünf Schritten:

1. Standort der Gegenstation aus dem Locator, sonst aus `cty.dat`; von dort der Großkreis zur eigenen Station.
2. Der Weg wird in Sprünge von höchstens **3500 km** geteilt; Kontrollpunkt ist die Mitte jedes Sprungs.
3. Je Kontrollpunkt sagen die Konturen, zwischen welchen zwei Stufen foF2 und MUF liegen. Auf die tatsächliche Sprungweite umgerechnet wird mit dem Sekantengesetz, bei fest angenommener Reflexionshöhe von **300 km**.
4. Die **Strecken-MUF ist das Minimum** über alle Kontrollpunkte – ein Sprung, der nicht zurückkommt, macht den ganzen Weg zu.
5. Die Ampel: **offen** bis $0,85 \times \text{MUF}$ (die klassische *frequency of optimum traffic*), **grenzwertig** bis zur MUF, darüber **zu**. Gerechnet wird für die Bänder von 1,8 bis 54 MHz, ohne 8 m.

Was das Verfahren nicht kann, und zwar grundsätzlich, nicht aus Nachlässigkeit:

- **Keine Dämpfung, also keine LUF.** Es gibt kein D-Schicht-Modell; die Tageslichtmarke an 160/80/40 m ist eine Faustregel, kein Modell. Nach dem Verfahren allein wäre 80 m mittags immer offen.
- **Kein Rauschen, keine Antennen, keine Leistung.** Es sagt *die Ionosphäre trägt*, nicht *die Verbindung kommt zustande*.
- **Die Auflösung ist die der Konturstufen**, und die liegen auf den Bandfrequenzen. Für die Frage „trägt 20 m“ ist das genau richtig; eine MUF auf 0,1 MHz genau steht dort nicht.
- **Keine Vorhersage.** Alles bezieht sich auf jetzt. Für 24 Stunden voraus gibt es den HF Propagation Planner auf `prop.kc2g.com`.

Beteiligte Module, falls Sie in den Quellcode sehen: `solar.py` (Feed und Zwischenspeicher), `propagation.py` (der Monat an Messwerten), `muf.py` (Konturen, Stationswerte, Freigabe), `pathmuf.py` (die Rechnung aus den fünf Schritten oben), dazu `api/solar.py` und `api/muf.py` als Brücke zur Oberfläche.

4.7 Workflow 7 – Vorgaben für QSOs

Werkzeuge ▶ *Vorgaben für QSOs* verwaltet benannte Sätze von Startwerten – etwa „FT8 Digital“ mit Mode, Submode und Rapporten. Der eingeschaltete Satz füllt die leere Maske vor; überschreiben und löschen geht dort jederzeit. Felder der Gegenstation sind bewusst nicht dabei.

4.8 Workflow 8 – Anzeige anpassen

Werkzeuge ▶ *Anzeige* (gilt nur für dieses Fenster und übersteht den Neustart):

Einstellung	Wirkung
QRZ.com-Profile	In einem LogRunner-Fenster oder im Browser des Systems öffnen
Bandplan	IARU-Region 1, 2 oder 3 – teilt die Spalten der Band-Map ein

Menüband, untere Statusleiste und Hilfetexte standen hier bis 0.9.3b4 und hängen jetzt allein am **Rechtsklick** – siehe *Markieren, Kopieren und das Kontextmenü*.

4.9 Workflow 9 – Alle QSOs löschen

Werkzeuge ▶ Alle QSOs löschen

 **Achtung:** Das löscht den **gesamten** Log. Alle QSOs samt Zusatzfeldern und Sync-Status werden aus der Datenbank entfernt. In der App lässt sich das nicht rückgängig machen.

Was geschieht:

- Vorher wird automatisch eine vollständige Sicherung der Datenbank in `~/LogRunner/backups/` angelegt.
Lässt sich diese Sicherung nicht schreiben, wird nichts gelöscht.
- Erhalten bleiben: Standorte, Wavelog-Zugang, Sprache und Thema.
- Der Abruf-Zeiger für Wavelog wird zurückgesetzt, damit *Jetzt abholen* die QSOs bei Bedarf wieder von dort holen kann.
- Zur Bestätigung ist die genaue Zahl der vorhandenen QSOs einzutippen. Stimmt sie nicht, wird nichts gelöscht.

Wiederherstellen: LogRunner schließen, die gewünschte Datei aus `~/LogRunner/backups/` nach `~/LogRunner/logrunner.sqlite` zurückkopieren, LogRunner wieder starten.

5. Einstellungen & Konfiguration

5.1 Wo die Einstellungen liegen

Alle Einstellungen stehen in `~/LogRunner/config.toml`. Die Datei wird beim ersten Start mit den Voreinstellungen angelegt und danach vom Programm gepflegt; jede Änderung in einem Dialog landet dort. Sie lässt sich auch von Hand bearbeiten – dann bei geschlossenem Programm.

Jedes **Begleitfenster** hat seine eigene `config.toml` unter `~/LogRunner/clients/<Name>/`. So merkt sich jedes Fenster seine eigene Größe, seine eigene Ecke und seinen eigenen Bereich.

⚠ Achtung: In `config.toml` stehen der Wavelog-API-Schlüssel sowie die Passwörter für QRZ.com und HamQTH im **Klartext**. Eine Verschlüsselung gibt es in dieser Fassung nicht. Wer die Datei lesen kann, liest auch die Zugangsdaten.

5.2 Die Abschnitte von `config.toml`

Abschnitt	Inhalt
[general]	Sprache, Thema, Ansicht, Betriebsart, Spalten, Kartenebenen, Bandplanregion und weitere Fensterangelegenheiten
[window]	Größe, Stelle auf dem Desktop und ob das Fenster maximiert war (<i>vom Programm geschrieben</i>)
[wavelog]	Adresse, API-Schlüssel, Standortnummer, Abruf-Zeiger je Standort
[lookup]	QRZ.com, HamQTH, callook.info, Positionen der Spots, Zeitlimit
[wsjtx]	Ein/aus, Adresse, Port
[cat]	Ein/aus, Anbindung, Abfragetakt, Richtungsschalter, Wasserfall, je Anbindung ein Unterabschnitt
[livelink]	Ein/aus, Rolle, Hub-Adresse, Port, PC-Name, mDNS, Maske, Bandzuweisung, Autostart der Begleitfenster
[capture]	Ein/aus, Gerät, Format, Ausgabe, Netzfreigabe, Schlüssel, ffmpeg-Pfad, Bildquelle
[solar]	Ein/aus des stündlichen Abrufs der Sonnenzahlen
[muf]	Ein/aus des Abrufs der MUF-Konturen von prop.kc2g.com
[dxcluster]	Ein/aus, Autoverbindung, Login-Rufzeichen, Puffer, Ansicht, aufgeklappter Bereich, drei Verfallsfristen, Skimmer, Filter, Filtersätze, Alarmer, Serverliste

5.3 Wichtige Einzeloptionen

Schlüssel	Werte	Voreinstellung	Bedeutung
general.language	de, en, es, fr, it, pt	en	Sprache der Oberfläche
general.theme	dark, light	dark	Farbthema
general.view	detail, compact	detail	Wie viel Fenster gezeigt wird
general.entry_mode	log, contest	log	Wofür die Eingabemaske steht
general.menu_bar	true / false	true	Menüband oben (Kopfzeile mit den Menüs)

Schlüssel	Werte	Voreinstellung	Bedeutung
general.status_bar	true / false	true	Untere Statusleiste
general.show_hints	true / false	true	Erklärtexte unter Feldern
general.prefill	true / false	false	Band/Mode/Submode/Frequenz übernehmen
general.freq_unit	Hz, kHz, MHz, GHz	MHz	Anzeigeeinheit der Frequenzfelder (gespeichert wird stets MHz)
general.qrz_open_in	window, browser	window	Wo ein QRZ-Profil aufgeht
general.bandmap_region	1, 2, 3	1	IARU-Bandplan für die Band-Map
general.scp_auto_update	true / false	true	SCP-Liste selbst aktuell halten
general.frontend_source	server, file	server	Vom Programm gesetzt – siehe Abschnitt 6
wavelog.enabled	true / false	false	Wavelog-Abgleich
lookup.timeout_s	Zahl	4.0	Zeitlimit einer Rufzeichenabfrage
wsjtx.host / .port	Adresse / Zahl	127.0.0.1 / 2237	Wo auf WSJT-X gehört wird
cat.backend	icom, rigctld, omnirig	icom	Anbindung an den Transceiver
cat.poll_ms	Zahl	300	Wie oft das Gerät gefragt wird
cat.qsy_on_spot	true / false	true	Angeklickte Meldung ans Gerät geben
cat.adopt_from_rig	true / false	true	Gerät → Maske
cat.all_bands	true / false	false	Alle Bandtasten zeigen statt der Bänder, die das Gerät meldet
livelink.port	Zahl	8765	Port des Hubs
livelink.autostart	true / false	false	Begleitfenster beim Start des Hubs mit hochfahren
livelink.autostart_presets	Liste von {name, shortcuts}	leer	Die Fenster-Presets; shortcuts sind Dateinamen
livelink.autostart_preset	Name	leer	Welches Preset der Autostart startet; leer = keines
livelink.autostart_shortcuts	Liste von Dateinamen	leer	Die Fenster des Autostart-Presets, mitgeschrieben für ältere Versionen
capture.port	Zahl	8770	Port des Bildservers
dxcluster.max_spots	50–5000	500	Meldungen im Puffer
dxcluster.show_skimmer	true / false	false	RBN-Meldungen annehmen
dxcluster.*_retention_minutes	0, 5, 10, 15, 30, 60	0 (nie)	Verfallszeit für Band-Map, Karte und Wasserfall – je eine

5.4 Umgebungsvariable

Variable	Wirkung
LOGRUNNER_HOME	Legt den Datenordner fest. Ohne sie gilt ~/LogRunner .

5.5 Kommandozeile

Unbekannte Optionen werden übergangen und protokolliert, nicht bestraft – ein Fensterprogramm ohne Konsole könnte einen Nutzungshinweis nirgends hinschreiben.

Option	Wirkung
--client [Name]	Startet ein Begleitfenster mit eigenem Datenverzeichnis unter ~/LogRunner/clients/<Name>/ . Ohne Namen wird der nächste freie vergeben.
--mask <Kennung>	Welchen Bereich dieses Fenster allein zeigt (siehe Tabelle in Abschnitt 4.2)

Beispiel:

```
LogRunner.exe --client Bandmap --mask cluster:map
```

5.6 Tastenkombinationen

Taste	Wo	Wirkung
Esc	offener Werkzeugdialog	Dialog schließen, Cursor zurück ins Rufzeichenfeld
Esc	Contest-Zeile	Eingabe verwerfen, Cursor zurück aufs Rufzeichen
Esc	Meldeformular des Clusters	Formular schließen, ohne zu melden
Esc	offenes Menü	Menü schließen
Enter	Eingabemaske	QSO loggen (in der Contest-Zeile aus jedem Feld)
Strg + Alt + Enter	Contest-Zeile	Dublette trotzdem loggen
^ (links neben der 1)	ohne Fokus, im Rufzeichenfeld, in der Contest-Zeile auch in <i>Nr. empf.</i>	Sendeumschaltung (PTT) – am eigenen Gerät, oder über die Bandzuweisung am Gerät eines anderen PCs
Leertaste	Contest-Zeile: Rufzeichen oder <i>Nr. empf.</i>	Zwischen Rufzeichen und <i>Nr. empf.</i> hin- und herspringen. Zählt der Contest keine Nummern, springt sie stattdessen ins Feld <i>Austausch empfangen</i>
Tab	Eingabemaske	Zum nächsten Feld; die eigene laufende Nummer wird übersprungen
↓ / ↑	Menüknopf	Menü öffnen und darin navigieren
↑ / ↓	Teiler im Wasserfall	Aufteilung zwischen Kurve und Wasserfall feinjustieren
Doppelklick	Teiler im Wasserfall	Aufteilung zurücksetzen
Doppelklick	Rufzeichenfeld, Log-Zeile, DX-Meldung, Karte	QRZ.com-Profil öffnen
Einfachklick	DX-Meldung, Log-Eintrag auf der Karte	In die Eingabemaske übernehmen
Mausrad	Karte	Zoomen

 **Achtung:** Die ^-Taste schaltet den Sender um. Sie wirkt deshalb ausdrücklich **nur** dann, wenn nichts fokussiert ist, der Cursor im Rufzeichenfeld steht oder – in der Contest-Zeile – in *Nr. empf.* In Name-, QTH- und Kommentarfeldern bleibt ^ ein Zeichen. Erkannt wird die Taste an ihrer Lage links neben der 1, nicht am Zeichen darauf: auf einer US-Tastatur ist es die Taste mit dem ` , auf einer französischen die mit dem ². Ohne verbundene TRX-Steuerung geschieht gar nichts.

Bis 0.9.3b55 war es die Leertaste. Die gehört seitdem in der Contest-Zeile dem Sprung zwischen Rufzeichen und *Nr. empf.* – wie in jedem Contestlogger.

F1 bis F12 sind absichtlich nicht belegt. In jedem Contestlogger tragen diese Tasten die CW-Makros; LogRunner gibt sie frei, damit ein Keyer sie später unverändert übernehmen kann.

6. Fehlerbehebung (Troubleshooting)

6.1 Wo die Protokolle liegen

Datei	Inhalt
~/LogRunner/logs/logrunner.log	Das laufende Protokoll. Rollierend, höchstens 512 KB je Datei, zwei Sicherungen. Enthält am Anfang jedes Starts einen Umgebungsbericht (Version, Windows, Python, gepackt ja/nein, WebView2-Version, Anzeige-Engine).
~/LogRunner/logs/start.log	Nur Startfehler – die Fälle, in denen es nie zu einem Fenster kam.
%TEMP%\logrunner-start.log	Notfallort, falls sich nicht einmal der Datenordner anlegen lässt.

Hilfe ▶ Über zeigt dieselben Systemangaben und legt sie auf Knopfdruck in die Zwischenablage – diese Angaben gehören in jeden Fehlerbericht.

Ausführlicheres Protokoll für eine Fehlersuche. Normalerweise schreibt LogRunner nur, was von Belang ist. Wer einen Fehler nachstellt und mehr sehen will, startet einmal mit gesetzter Umgebungsvariable `LOGRUNNER_DEBUG=1` – dann steht auch der Kleinkram im Protokoll:

```
$env:LOGRUNNER_DEBUG = 1
& "C:\LogRunner\LogRunner\LogRunner.exe"
```

Auf dem Pi entsprechend `LOGRUNNER_DEBUG=1 ~/Programme/LogRunner/LogRunner`. Das Protokoll wird dadurch deutlich länger; für den Alltag gehört der Schalter wieder weg.

Auch die Oberfläche schreibt dann mit. Wer zum Beispiel eine Cluster-Meldung anklickt und der Transceiver folgt nicht, findet im Protokoll eine Zeile `Oberf1aeche [qsyToSpot]: ...` – sie nennt den Weg, den der Klick genommen hat (an einen anderen Rechner geschickt, kein eigenes Gerät, *Auf Spot-Frequenz gehen* ausgeschaltet oder hier gedreht) und die Werte, aus denen sich das ergab. Das ist der Unterschied zwischen einer Vermutung und einem Befund: ohne diese Zeile sehen drei der vier Wege von außen gleich aus – es passiert nichts.

6.2 Startprobleme

Meldung	Bedeutung und Abhilfe
„Auf diesem Rechner fehlt die Microsoft Edge WebView2 Runtime.“	Ohne sie würde pywebview auf die alte Internet-Explorer-Anzeige zurückfallen, in der die Oberfläche nicht läuft. Den <i>Evergreen Standalone Installer (x64)</i> für WebView2 installieren und neu starten.
„Die Microsoft Edge WebView2 Runtime ... ist Version n. LogRunner braucht mindestens Version 111.“	Eine vorhandene, aber zu alte Runtime. Sie ist schlimmer als gar keine: alles startet, die Oberfläche bleibt aber ein schwarzes Rechteck. Derselbe Installer bringt sie auf Stand.
„LogRunner läuft bereits (Lock: ...)“	Es läuft bereits eine Instanz auf demselben Datenverzeichnis. Zwei Prozesse auf derselben SQLite-Datei sind die eine Art, ein Logbuch zu zerstören, gegen die kein Backup schnell genug ist. Das laufende Fenster suchen – gegebenenfalls im Task-Manager nach <code>LogRunner.exe</code> . Eine Sperrdatei aus einem Absturz hält niemanden auf.
„Datenbank beschädigt: ... Bitte aus einem Backup in ... wiederherstellen.“	Die Integritätsprüfung der SQLite-Datei ist fehlgeschlagen. LogRunner schließen, die jüngste Datei aus <code>~/LogRunner/backups/</code> nach <code>~/LogRunner/logrunner.sqlite</code> kopieren, neu starten.

Meldung	Bedeutung und Abhilfe
„Die Einstellungsdatei lässt sich nicht lesen: ...“	<code>config.toml</code> wurde von Hand bearbeitet und ist nun fehlerhaft. Die genannte Stelle berichtigen – oder die Datei umbenennen; dann legt LogRunner eine neue mit den Voreinstellungen an. Die Zugangsdaten müssen dann neu eingetragen werden.
„Der Web-Teil des Programms fehlt: ... ist nicht da. Bitte das Archiv vollständig entpacken.“	Der Ordner <code>_internal</code> fehlt oder ist unvollständig. Das ZIP-Archiv erneut und vollständig entpacken.
Das Fenster bleibt leer oder schwarz	LogRunner erkennt das selbst: Es fragt die Seite, wartet bis zu 25 Sekunden und versucht danach automatisch einen zweiten Weg – die Dateien direkt aus dem Ordner statt über den lokalen Webserver. Gelingt das, wird <code>general.frontend_source = "file"</code> in <code>config.toml</code> vermerkt, und der nächste Start geht gleich diesen Weg. Gelingt es nicht, erscheint ein Meldungsfenster mit Anzeige-Motor, Seitenzustand und gemeldeten Fehlern. Häufigste Ursachen: eine zu alte WebView2-Runtime oder ein Rechner, der dem Programm die eigenen Dateien nicht lesen lässt (Virens Scanner, Gruppenrichtlinie).
Das Startfenster fragt bei jedem Start wieder nach der Lizenz	Der Vermerk <code>~/LogRunner/license.json</code> ließ sich nicht schreiben – ein schreibgeschütztes Datenverzeichnis, ein Stick ohne Schreibrecht. Im Protokoll steht eine Zeile dazu. Das Programm läuft trotzdem; nur die Frage kommt wieder.
„Die Datei LICENSE.txt konnte nicht gefunden werden.“	Sie gehört neben <code>LogRunner.exe</code> und fehlt dort. Das Archiv erneut vollständig entpacken. Die Frage nach der Zustimmung wird trotzdem gestellt – der Vertrag gilt auch ohne die Datei.
Zwei LogRunner-Fenster zeigen dasselbe, obwohl eines neuer ist	Bis 0.9.3b2 teilten sich mehrere Fenster auf einem PC einen Webserver – das zweite lud die Oberfläche des ersten. Seither bekommt jedes Fenster einen eigenen Port; im Protokoll steht <i>Oberfläche wird auf Port N ausgeliefert</i> .
Nach einem Neustart ist die QRZ-Anmeldung wieder weg	Der Ordner <code>~/LogRunner/data/webview/</code> ließ sich nicht beschreiben; die Anzeigebibliothek läuft dann im privaten Modus. Im Protokoll steht eine Zeile dazu. Sonst ist nichts betroffen.
Es passiert beim Doppelklick gar nichts	Der gepackte Build hat keine Konsole. Nachsehen in <code>~/LogRunner/logs/start.log</code> .

6.3 Logbuch und Daten

Situation	Bedeutung und Abhilfe
„Bestätigung passt nicht: n statt m.“	Beim Löschen des gesamten Logs wurde die falsche Zahl eingetippt – oder das Fenster zeigte einen veralteten Stand. Es wurde nichts gelöscht.
„Sicherung konnte nicht angelegt werden – es wurde nichts gelöscht.“	Volle Platte oder ein schreibgeschützter Ordner. Ohne Sicherheitsnetz wird nicht gelöscht.
„Nicht geloggt – im Rufzeichenfeld steht ein Suchmuster“	Im Rufzeichenfeld steht ein <code>?</code> oder <code>*</code> . Ein Rufzeichen, das niemand hält, ginge an Wavelog, LoTW und in die Cabrillo-Datei und ließe sich von dort nicht zurückholen.
„Nicht geloggt – Strg+Alt+Enter loggt trotzdem“	Contestmodus, und die Station steht in diesem Contest bereits im Log.
Fehlerhafte Datensätze im Importbericht	Zeilen, die der ADIF-Parser nicht lesen konnte. Der Rest der Datei wurde eingelesen; die Sicherung vor dem Import steht im selben Bericht.
Alte QSOs haben keine DXCC-Nummer	Neu geloggte QSOs bekommen die Entity-Nummer automatisch. Für Bestandsdaten aus einem Import gilt das nicht rückwirkend.

6.4 Wavelog-Abgleich

Meldung	Bedeutung und Abhilfe
„Keine Verbindung zu Wavelog – die QSOs bleiben in der Warteschlange ...“	Kein Fehler im engeren Sinn. Die QSOs bleiben stehen und gehen beim nächsten <i>Jetzt hochladen</i> mit. Von selbst passiert das nicht – seit 0.9.3b45 gleicht LogRunner nur ab, wenn der Knopf gedrückt wird.
„Dem Token fehlen Scopes: ...“	Verbindung und Stationsliste funktionieren, das Hochladen wird fehlschlagen. Das Token in Wavelog um <code>qso:read</code> und <code>qso:write</code> ergänzen.
„Dafür wird API v2 gebraucht – ein Token mit dem Präfix <code>w12_</code> .“	<i>Änderungen senden</i> geht nur mit API v2; die alte API kann ein bereits hochgeladenes QSO nicht mehr ändern.
„n QSOs nicht zugeordnet: Wavelog hat dazu keinen eindeutigen Eintrag.“	Dort gelöscht, oder es stehen zwei gleiche.
„n QSOs warten weiter: ihr Standort hat kein Wavelog-Logbuch zugeordnet.“	Unter <i>Stationsstandorte</i> dem betroffenen Standort einen Wavelog-Standort zuweisen.
„Das Konto hat n Standorte; die übrigen werden erst abgeholt, wenn ...“	Dasselbe in der Gegenrichtung.

6.5 Rufzeichen-Abfrage

Der Testknopf im Panel nennt den Grund im Klartext:

Grund	Bedeutung
<i>Es ist kein Dienst eingeschaltet.</i>	Alle drei Dienste sind aus
<i>Kein Rufzeichen eingegeben.</i>	Das Testfeld ist leer
<i>Kein US-Rufzeichen – und callook.info ist der einzige eingeschaltete Dienst.</i>	callook kennt nur K, N, W und AA–AL
<i>Keiner der gefragten Dienste kennt dieses Rufzeichen.</i>	Antwort erhalten, aber ohne Treffer
<i>Der Dienst ist nicht erreichbar.</i>	Kein Netz oder Zeitlimit überschritten
<i>Benutzername oder Passwort wurden abgelehnt.</i>	Zugangsdaten prüfen
<i>Die Abfrage ist fehlgeschlagen.</i>	Unerwartete Antwort des Dienstes

Fehlt bei QRZ.com die Hälfte der Felder, liegt das meist am Abonnement: den vollständigen Datensatz gibt die XML-Schnittstelle nur an Konten mit *XML Logbook Data* heraus. Die Begründung erscheint bei der Testabfrage.

6.6 DX-Cluster

Meldung	Bedeutung und Abhilfe
„Kein Login-Rufzeichen: weder hier noch beim Stationsstandort ist eines hinterlegt.“	Ohne Rufzeichen weist jeder Knoten die Verbindung ab. Eines im Cluster-Dialog oder im aktiven Stationsstandort eintragen.
„Kein Server eingetragen“	In den Cluster-Einstellungen mindestens einen Knoten anhängen.
„Nicht verbunden.“	Der Knoten ist nicht erreichbar oder hat die Verbindung getrennt. Erneute Versuche laufen im Hintergrund; der Programmstart wird davon nie aufgehalten.

Meldung	Bedeutung und Abhilfe
„Alle Meldungen sind älter als die eingestellte Zeit.“	Die Verfallszeit greift. Die Tabelle zeigt die Meldungen weiterhin – sie ist die Aufzeichnung, Band-Map, Karte und Wasserfall zeigen, wo <i>jetzt</i> etwas zu hören ist.
„Die Statistik-Datenbank lässt sich nicht öffnen.“	~/LogRunner/data/dxcluster.sqlite ist beschädigt oder schreibgeschützt. Nur das Diagramm bleibt leer; die Meldungsliste ist nicht betroffen. Die Datei lässt sich gefahrlos löschen, sie wird neu aufgebaut.
Die Liste läuft mit Maschinenmeldungen voll	Skimmer-Meldungen (RBN) sind eingeschaltet. Ausschalten – oder den Puffer vergrößern und mit dem Spotter-Filter *-# gezielt arbeiten.

6.7 TRX-Steuerung und Wasserfall

Situation	Bedeutung und Abhilfe
Die Schnittstelle lässt sich nicht öffnen	Ein anderes Programm hält sie – häufig ein noch laufendes rigctld, Omnirig oder eine Fernsteuerungssoftware. Auch ein abgestürzter Vorgängerprozess kann den COM-Port noch halten; LogRunner gibt ihn beim Beenden ausdrücklich zurück.
„Sendezustand nicht abfragbar“	Das Gerät kennt Befehl 1Ch nicht (Geräte älter als etwa 2005). Dann <i>Sendezustand abfragen</i> abschalten, sonst wartet jede Abfrage vergeblich.
Verbunden, aber nichts kommt an (Icom)	CI-V-Adresse und Baudrate im Gerätemenü prüfen. Bei Geräten mit eingebautem USB muss CI-V USB Port auf <i>Unlink from REMOTE</i> stehen. CI-V Transceiver eingeschaltet lassen.
„Dieser Transceiver liefert über CI-V kein Spektrum.“	Kein Bandscope, oder die Verbindung läuft über den [REMOTE] -Anschluss. Der Wasserfall geht nur über USB oder LAN.
„Der Transceiver steht außerhalb des angezeigten Bereichs.“	VFO und dargestellter Ausschnitt passen nicht zusammen.
Ein Klick auf eine Meldung stellt das Gerät nicht	Angeklickte DX-Meldung an den Transceiver geben ist aus.
Im Wasserfall stehen keine oder kaum DX-Meldungen	Der abgebildete Ausschnitt ist zu schmal. In CENTER ist er $VFO \pm Span$ – bei ± 5 kHz liegt fast jede Meldung außerhalb und wird deshalb nicht gezeichnet; anklicken lässt sich dort nur, was auch im Bild steht. Span erweitern oder das Bandscope auf FIX mit passenden Rändern stellen.
Die Leiste <i>Gerät bedienen</i> ist leer oder fehlt	Sie gibt es nur bei Icom CI-V. Steht dort <i>Dieser Transceiver gibt keine dieser Einstellungen über CI-V heraus</i> , hat das Gerät jeden der Befehle abgelehnt – dann CI-V-Adresse und Baudrate prüfen.
Die Anzeige friert ein, die Remote Utility zeigt aber <i>connected</i>	Der virtuelle COM-Port trägt nicht, was über ihn laufen soll – typisch nach dem Hochsetzen der Baudrate bei eingeschaltetem Wasserfall und kurzem Abfragetakt. Schrittfolge siehe unten.
Im Protokoll steht seit Minuten nichts Neues	Kein gutes Zeichen und kein schlechtes: LogRunner schreibt einen sich wiederholenden Fehler nur beim ersten Mal ins Protokoll, sonst stünden dort zweihundert gleiche Zeilen. Es versucht weiterhin alle fünf Sekunden. Was gerade gilt, steht im CAT-Panel, nicht in der Datei.

Wenn die Verbindung über die Icom Remote Utility einfriert, in dieser Reihenfolge:

1. In *Werkzeuge* ▶ *TRX-Steuerung* die CAT-Verbindung **abschalten**. Damit hören die Versuche im Fünf-Sekunden-Takt auf, und LogRunner gibt die Schnittstelle frei.
2. Die **Icom Remote Utility beenden und neu starten**. Das setzt den virtuellen COM-Port neu auf; er kann bis dahin sogar ganz aus der Liste der Schnittstellen verschwunden sein.
3. Den **Abfragetakt auf 300 ms** stellen, CAT wieder einschalten und nachsehen, ob die Frequenz mitläuft.

4. Erst danach den **Wasserfall** wieder zuschalten – so ist zu erkennen, welcher der beiden Teile die Leitung überfordert hat.

Im Protokoll (~/LogRunner/logs/logrunner.log) sind drei Meldungen zu unterscheiden, sie stehen für drei verschiedene Zustände:

Meldung	Zustand
Keine Antwort von CI-V-Adresse ...h an COM...	Die Schnittstelle ist offen, die Gegenseite schweigt – falsche Adresse oder Baudrate, oder der Weg dahin trägt nicht mehr.
... lässt sich nicht öffnen: Das System kann die angegebene Datei nicht finden	Die Schnittstelle gibt es gerade nicht . Bei einem virtuellen Port heißt das: das Programm dahinter hat ihn abgemeldet.
... lässt sich nicht öffnen: Zugriff verweigert	Es gibt sie, aber ein anderes Programm hält sie – auch ein abgestürzter Vorgängerprozess von LogRunner selbst. Hilft ein Neustart der Remote Utility nicht, hilft ein Neustart des Rechners.

6.8 WSJT-X-Übernahme

Situation	Bedeutung und Abhilfe
„WSJT-X-Übernahme konnte nicht starten“ (Protokoll)	Der UDP-Port ist belegt – meist von JTAlert, GridTracker oder einer noch laufenden LogRunner-Instanz. Lösung: in allen beteiligten Programmen dieselbe Multicast-Adresse (z. B. 224.0.0.1) verwenden.
Es kommt nichts an	In WSJT-X unter <i>Datei</i> ▶ <i>Einstellungen</i> ▶ <i>Reporting</i> prüfen, dass „UDP Server“ auf dieselbe Adresse und denselben Port zeigt. „Accept UDP requests“ ist nicht nötig – LogRunner hört nur zu.
Rufzeichen oder Locator weichen ab	Eigenes Rufzeichen und Locator kommen aus dem aktiven Stationsstandort, nicht aus WSJT-X. Weichen sie ab, steht das in der Ereignisliste; geloggt wird unter dem Standort.

6.9 Bildaufnahme

Meldung	Bedeutung und Abhilfe
„ffmpeg ist auf diesem Rechner nicht vorhanden.“	Der Knopf <i>ffmpeg holen</i> lädt es (rund 110 MB, einmalig). Unter Linux/macOS über die Paketverwaltung installieren und den Pfad eintragen.
„Keine Kamera gefunden. Steckt die Capture-Karte?“	Karte prüfen, dann <i>Neu suchen</i> .
„Diese Kamera liefert kein MJPEG – das Bild wird umgerechnet.“	Kein Fehler, aber Rechenzeit. Falls möglich eine MJPEG-Betriebsart wählen.
„Verbindung zum Bild abgerissen“	Der Bildstrom ist abgebrochen. <i>Erneut versuchen</i> . Der Zähler <i>n Neustarts</i> zeigt, wie oft das schon geschah.
Das Bild ruckelt im WLAN	Auf <i>Neu berechnen</i> umstellen und vor allem Breite und ausgehende Bildrate reduzieren – das wirkt stärker als die Qualitätsstufe.

6.10 LiveLink

Situation	Bedeutung und Abhilfe
Der Hub wird nicht gefunden	mDNS wird im Netz geschluckt. Die Hub-Adresse von Hand eintragen – sie schlägt die Suche. Ersatzweise gilt die zuletzt gefundene Adresse.

Situation	Bedeutung und Abhilfe
Die Firewall fragt	Beim ersten Start eines Hubs normal. Unter Linux den Port freigeben (<code>ufw allow 8765/tcp</code>) und den <code>avahi-daemon</code> bereitstellen.
Auf dem Client sind Knöpfe ausgegraut	Ein spiegelndes Fenster loggt nicht selbst und hält keine Cluster-Verbindung. Geloggt und verbunden wird auf dem Hub; Filter und Alarme stellt jedes Fenster für sich ein.
Ein Begleitfenster zeigt ein leeres Fenster	Die eingetragene Maske ist unbekannt. LogRunner fällt dann auf das ganze Fenster zurück; gültige Kennungen stehen in Abschnitt 4.2.

6.11 Wenn gar nichts hilft

1. LogRunner schließen.
2. `~/LogRunner/logs/logrunner.log` und `start.log` sichern.
3. In *Hilfe* ▶ *Über* die Systemangaben in die Zwischenablage legen (beim nächsten Start) und beides dem Fehlerbericht beilegen.
4. Zum Zurücksetzen der Oberfläche genügt es, `config.toml` umzubenennen – das Logbuch selbst bleibt davon unberührt. **Die Zugangsdaten müssen danach neu eingetragen werden.**

Anhang: Hilfe und Unterstützung

Wenn dir LogRunner die Arbeit im Shack oder unterwegs erleichtert, freue ich mich über einen virtuellen Kaffee. Dein Support fließt direkt in die Weiterentwicklung, neue Features und Prototypen.

- Ko-fi: <https://ko-fi.com/dl1egl>
- PayPal: https://www.paypal.com/donate/?hosted_button_id=3CPP6CJSEUTG

Beides steht auch im Programm: *Hilfe* ▾ → *Über LogRunner*, Abschnitt *Hilfe / Unterstützung*. Die Knöpfe dort öffnen die Seite im Browser des Systems – dort sitzen Anmeldung und Zahlungsmittel.

Anhang: Daten und Lizenzen

LogRunner selbst wird unter einem Endnutzer-Lizenzvertrag überlassen; er liegt als `LICENSE.txt` neben `LogRunner.exe` und wird beim ersten Start und nach jeder neuen Version einmal im Startfenster vorgelegt (siehe 2.7). Die Bedingungen der enthaltenen Software und Daten Dritter stehen in `THIRD-PARTY-NOTICES.txt` daneben.

LogRunner bringt folgende Daten mit:

Datensatz	Quelle und Lizenz
Küstenlinien, Grenzen, Flüsse	Natural Earth 1:10m (gemeinfrei)
CQ- und ITU-Zonen	<code>hamradio-zones-geojson</code> , MIT, © 2024 HB9HIL
Ortsdaten	GeoNames, CC BY 4.0
DXCC-Präfixe (<code>cty.dat</code>)	AD1C, country-files.com

POTA-, SOTA-, IOTA- und Super-Check-Partial-Listen werden nicht mitgeliefert, sondern auf Knopfdruck von der jeweiligen Quelle geholt.