MobaLedLib Studio – Das Programm aktualisieren

Seitenstand 1.05 (2026-11-24), gilt für Version 5.12

Neue Versionen bringen neue Funktionen und beheben Fehler. Das Programm **aktualisiert sich selbst**: Du musst nichts von Hand kopieren. **Deine Tabellen und Einstellungen bleiben erhalten**, weil sie nicht im Programmordner liegen, sondern im **Datenordner** (siehe unten).

Das Wichtigste in drei Schritten

1. Öffne das Menü **Hilfe → Update MobaLedLib Studio …**.
2. Prüfe die Liste und klicke auf **Aktualisieren**.
3. Klicke danach auf **Neu starten** und lade die Seite im Browser mit **Strg + F5** neu.

Beende das Programm dafür **nicht** von Hand: Du aktualisierst im laufenden Programm.

Schritt für Schritt

1. **Hilfe → Update MobaLedLib Studio …** öffnet ein Fenster. Das Programm lädt die **neueste fertige Version** von GitHub und vergleicht **jede Datei** mit deiner Installation.
2. Das Fenster zeigt, **welche Version** gefunden wurde und **was sich ändert**: neue, geänderte und zu löschende Dateien, jeweils mit der Versionsnummer. Es passiert **noch nichts**, bis du **Aktualisieren** klickst.
3. Nach dem Aktualisieren läuft noch die **alte Version**. **Neu starten** sichert deine Tabellen und startet das Programm neu. Der offene Reiter lädt sich selbst neu. Auf dem **Raspberry Pi** startet der Dienst von selbst neu.
4. Lade die Seite im Browser mit **Strg + F5** neu. Die **Versionsnummer** steht unter dem Titel der Oberfläche. Oben darf **keine rote Meldung** stehen.



Ist das Aktualisieren sicher?

- **Nur Geändertes wird geschrieben.** Das sind meist wenige Dateien statt des ganzen Ordners. Gleiche Dateien bleiben unberührt.
- **Alles oder nichts:** Vor dem Schreiben werden die ersetzten Dateien gesichert. Schlägt ein Schritt fehl (zum Beispiel weil ein Sicherungsprogramm eine Datei sperrt), stellt das Programm **sofort den alten Stand** wieder her und meldet es.
- **Bricht es mittendrin ab** (Stromausfall, Programm beendet), macht das Programm das Update **beim nächsten Start selbst rückgängig**. Oben erscheint ein orange Hinweis, deine Tabellen und Einstellungen sind nicht betroffen. Versuche das Update später noch einmal.
- **Eigene Dateien** im Programmordner bleiben immer erhalten. Gelöscht werden nur Dateien,

die ein früheres Update installiert hat und die in der neuen Version fehlen.
- Das Programm lädt **nur über HTTPS** und installiert nur, wenn das Paket **vollständig** ist und zu seiner Prüfliste passt. Sonst bricht es ab, **ohne etwas zu verändern**.
- **Sicherungskopien:** Ersetzte und gelöschte Dateien liegen vorher im Datenordner unter `update_backup/<Datum-Uhrzeit>/` (die letzten fünf). Willst du eine Datei zurückholen, kopiere sie in den Ordner **python** zurück.

Wo liegt der Datenordner?

Dort liegen deine **Einstellungen**, die **Sicherung der Tabellen**, das **Passwort** und die **Protokolle**. Er liegt **außerhalb** des Programmordners.

System	Datenordner
Windows	`%APPDATA%\webMobaLedLib` (das ist `C:\Users\<Name>\AppData\Roaming\webMobaLedLib`)
macOS	`~/Library/Application Support/webMobaLedLib`
Linux, Raspberry Pi	`~/.local/share/webMobaLedLib` (beim Pi für den Benutzer, unter dem der Dienst läuft)

Vor einem größeren Update sichern: Speichere deine Tabelle mit dem Menü **Datei** (die Datei landet als Download auf dem Rechner, an dem du sitzt) und kopiere den Datenordner.

Besondere Meldungen im Update-Fenster

Meldung	Was sie bedeutet und was du tust
Rot: **„Achtung: Das Update löscht … Dateien (mehr als 20) …“**	Die neue Version enthält vieles nicht mehr, zum Beispiel nicht mehr benutzte Symbole. Das ist normal. **Abbrechen** ist der Standardknopf. Wenn du einverstanden bist, klicke **„Ja, … Dateien löschen und aktualisieren“**. Die gelöschten Dateien werden vorher gesichert.
Rot: **„ACHTUNG: Dieses ZIP hat eine ÄLTERE Version als die installierte!“**	Die gefundene Version ist **älter** als deine. Das willst du nur, wenn du **bewusst zurückgehen** möchtest (dann **„Trotzdem auf Version … zurückgehen“**). Sonst **Abbrechen**.
„… konnte nicht geladen werden“	Prüfe die Internetverbindung, die Firewall und einen eventuellen Proxy.
Bei einer Datei steht **„Nicht ersetzt“**	Ein anderes Programm (meist ein Virenscanner oder eine Datensicherung) hat die Datei gesperrt. Beende das Programm, halte das andere Programm kurz an und wiederhole das Update.
„Die Liste der Releases konnte nicht gelesen werden … CERTIFICATE_VERIFY_FAILED“ (macOS)	Python von python.org bringt auf dem Mac keine Zertifikate mit. **Ab Version 2.79** kümmert sich das Programm selbst darum. Bei einer älteren Version führst du einmal im Terminal open „`/Applications/Python 3.12/Install Certificates.command`“ aus (nimm den Ordner deiner Python-Version).

Meldung	Was sie bedeutet und was du tust
„nach dem Schreiben nicht lesbar“ (Windows, macOS), meist bei einer **alten** Version	Siehe den nächsten Abschnitt. Es ist **nichts verändert**, das Programm hat den alten Stand wiederhergestellt.
„keine Prüfliste“, **„unvollständig“** oder **„würde … Dateien löschen“**	Das bereitgestellte Paket ist nicht das vollständige Release. Es wurde **nichts verändert**. Melde es im Forum, bis das richtige Paket da ist.

Update aus einer sehr alten Version (Windows und macOS)

Das Update führt der Programmcode der **installierten** Version aus. Versionen **bis 2.97** löschen am Ende alle Dateien, die das neue Paket nicht kennt, auch wenn dieselbe Datei dort nur in **anderer Schreibweise** steht (zum Beispiel `InfoB.png` und `infoB.png`, unter Windows und macOS ist das **dieselbe Datei**). Dann meldet das Update **„nach dem Schreiben nicht lesbar“** und stellt den alten Stand wieder her. **Ab Version 3.00** enthält das Paket die alten Schreibweisen zusätzlich, das Update klappt dann ohne Meldung. Erscheint sie trotzdem: Beende das Programm und entpacke das Paket **von Hand** (siehe Fehlersuche: Wenn nichts mehr geht). Danach ist das Update-Programm neu genug.

Besonderheiten der Systeme

- **Windows, macOS, Linux:** Haben sich die benötigten Python-Pakete geändert, aktualisiert das Programm sie selbst.
- **Raspberry Pi:** Der Dienst startet nach dem Update von selbst neu. Haben sich Dateien im Ordner **pi** geändert, führe einmal `sudo bash ~/mobaledlib/python/pi/install_pi.sh –no-check` aus (die installierte Dienstdatei ist eine Kopie).
- **Verknüpfung und Name:** Das Programm hieß ab Version 1.31 **webMobaLedLib** (vorher pyMobaLedLib) und heißt seit Version 5.09 **MobaLedLib Studio**. Nach einem Update über das Menü heißen Symbol und Starter noch „webMobaLedLib“. Das Installationsskript legt sie neu als **MobaLedLib Studio** an. Es entfernt dabei die Symbole mit den alten Namen (webMobaLedLib, pyMobaLedLib).

Für Tester: eine Testversion wählen

Mit **Hilfe → Update Testversion …** lädst du eine Version von GitHub, **die du selbst aussuchst**. Das können **Pre-Releases** sein, also Versionen, die noch nicht fertig geprüft sind. Das normale **Update MobaLedLib Studio …** lädt dagegen immer nur die letzte **fertige** Version.

1. Das Programm fragt zuerst, ob du **wirklich** eine Testversion installieren möchtest (Standardknopf **Abbrechen**). Dann klicke **Testversion suchen**.
2. Es zeigt eine **Auswahlliste der letzten 15 Versionen**, die **neueste zuerst und vorgewählt**. Jeder Eintrag nennt Name, **Pre-Release** oder **Release** und Datum. Die laufende Version trägt **installiert**, ältere tragen **ältere Version**.
3. **Version laden** lädt die gewählte (ein Klick, wenn du die neueste willst). **Abbrechen** schließt das Fenster, ohne etwas zu ändern.
4. Danach läuft alles **wie beim normalen Update**: Anzeige der Dateien, **Aktualisieren**, Sicherung, **Neu starten**.

5. Releases, deren Name oder Tag mit **test** beginnt (zum Beispiel *test1*), sind **Tests des Entwicklers**. Sie stehen **am Ende der Liste** (höchstens fünf), sind **nie vorgewählt** und tragen den Zusatz **Entwicklertest**. So installiert kein Tester sie aus Versehen; wer einen solchen Test braucht, wählt ihn gezielt aus.
6. Wählst du eine **ältere** Version, erscheint die rote Warnung (siehe oben) und der Standardknopf bleibt **Abbrechen**. Einstellungen und Dateien können sich beim Zurückgehen ändern; deine Tabellen werden vorher gesichert.

- Statt einer Version von GitHub kannst du auch einen **eigenen Link (https) zu einer ZIP-Datei** angeben oder eine **ZIP-Datei auf dem Rechner auswählen** (seit 3.95). Die Datei wird wie webmobaledlib.zip geprüft und installiert. Das geht auch, wenn GitHub nicht erreichbar ist.
- Ohne Netz, hinter einem Proxy oder bei zu vielen Abfragen (GitHub erlaubt 60 je Stunde) steht eine Meldung im Fenster. Es wird nichts verändert.

Das Programm startet nach einem Update nicht mehr?

Siehe Fehlersuche: Wenn nichts mehr geht. Dort steht auch das **Reparaturskript**, das fehlende oder beschädigte Dateien wiederherstellt.

Für Entwickler

In einer **Git-Arbeitskopie** ist das Update gesperrt, damit es die Dateien nicht mit dem Release überschreibt (`MLL_ALLOW_UPDATE=1` hebt die Sperre auf). `MLL_UPDATE_URL` richtet das Update auf eine andere ZIP-Adresse.

From:
https://wiki.mobaledlib.de/ - **MobaLedLib Wiki**

Permanent link:
https://wiki.mobaledlib.de/anleitungen/mobaledlibstudio/aktualisieren

Last update: **2026/11/24 17:06**