

# Plugin Upgrade Guide



## Für bestehende Plugins

Diese Zusammenfassung richtet sich an Entwickler, die Ihre Plugins von LoxBerry V0.2.3 auf LoxBerry V1.0 oder höher updaten.

## plugin.cfg

- Das Plugin-Interface muss auf 2.0 umgestellt werden.
- Ist nach der Installation ein Reboot erforderlich, kann dies in der plugin.cfg direkt als Parameter in Sektion [SYSTEM] als `REBOOT=1` gesetzt werden. (Ein bereits installiertes Plugin kann in Perl und PHP mit `reboot_required("Reboot-Grund");` einen Reboot anfordern - siehe Modul-Dokumentation).

## Allgemein

- Skripte, die nicht aus dem Webinterface geladen werden sollen, z.B. Skripte für Daemons, Restart von Services usw., sollen nicht mehr im webfrontend-Verzeichnis gespeichert werden (diese waren und wären damit für jeden per Webbrowser erreichbar. Dafür gibt es im Plugin nun das bin-Verzeichnis. Dateien im bin-Verzeichnis werden automatisch +x (executable) gesetzt.
- Das Verzeichnis /webfrontend/cgi umbenennen in /webfrontend/htmlauth. Entsprechend lokale Pfade im Script anpassen, am besten durch die Nutzung der Variable `$lbphtmlauthdir` (gesetzt durch LoxBerry::System). Die Web-URL bleibt gleich.
- `~/templates/system/<Sprache>/language.dat` ist **deprecated** und wird nur noch aus Legacy-Gründen mitgeliefert.
  - Plugins, die bisher außer eigenen Phrasen auch System-Phrasen aus dieser Datei verwendet haben, sollten ihr Plugin auf eigene Sprachphrasen, oder komplett auf das multilinguale System von LoxBerry umstellen.
- Die sprachspezifischen Header, Footer, sowie die Success- und Error-Seite in `~/templates/system/<Sprache>/`, konkret `header.html`, `footer.html`, `error.html` und `success.html` sind **deprecated** und werden aus Legacy-Gründen weiter bereitgestellt.
  - Diese Dateien werden in LoxBerry V1.0 sprachspezifisch generiert, unterstützen aber nicht alle Funktionen des Templatesystems.
  - In höheren Versionen könnte dies entfernt werden.
  - Daher sollte auf die LoxBerry-Funktionen `lheader` und `lfooter` umgestiegen werden.

## Preinstall / Postinstall

- Wurden mit `sed` die Verzeichnisse im Pre-/Postinstall ersetzt, kann dies nun entfallen. Stattdessen bietet LoxBerry die Ersetzung von Verzeichnissen automatisch mit den [hier gelisteten Konstanten](#) (ganz unten) an. Die Ersetzung erfolgt dabei auf alle Textdateien im Plugin, auch auf `php`, `pl`, `cgi`, `html` usw.
- `sed` hat zum Zeitpunkt von `postinstall` keine Rechte mehr auf das Daemon-File. Alle Änderungen am Daemon - wenn überhaupt erforderlich - müssen im `preinstall` erfolgen.

## Daemon

- Der Daemon läuft weiterhin als `root`. Das LoxBerry-Home-Directory ist als globale System-Environmentvariable `$LBHOMEDIR` verfügbar, sowie [weitere Variablen](#).
- Wenn der Daemon genutzt wurde, um Software zu installieren oder Einstellungen als `root` durchzuführen, sollten diese Routinen ins `postroot.sh` verschoben werden, welches direkt bei der Installation ausgeführt wird. Damit erspart man sich den Reboot und die Routinen werden nur einmal pro Installation ausgeführt.

## Webinterface in Perl

- Nach der Umstellung sollte `use LoxBerry::Web;` verwendet werden. Damit wird implizit auch `LoxBerry::System` geladen.
- Alle Routinen, die bisher zur Ermittlung des Installationspfades gedient haben, können entfallen. Stattdessen können direkt die Variablen von [Perl-Modul LoxBerry::System](#) verwendet werden.
- Alle Routinen, die bisher die Header und Footer von LoxBerry gelesen und angezeigt haben, können entfallen. Stattdessen kann direkt `lheader` und `lfooter` aus [Perl-Modul LoxBerry::Web](#) verwendet werden.

## Mehrsprachigkeit

Wenn bereits Sprachdateien im INI-Stil pro Sprache genutzt wurden, können diese einfach für das *Translate*-Widget angepasst werden, auch wenn nicht das Template-System von LoxBerry genutzt wird:

- Sprachen in den Ordner `template/lang`
- Englisch muss heißen z.B. `language_en.ini` (Translate sucht nach `*_en.ini`)
- Erzeugt wird von Translate dann z.B. `language_de.ini`
- In den eigenen Scripts die Pfade und Dateinamen entsprechend anpassen

Ab dann zeigt Translate im Plugin die englischen Sprachdateien an und erzeugt nach Benutzerauswahl die anderen Sprachen.