

LiveCode Server as a command line interpreter

www.Rozek.com > LiveCode Server> Installation as Command

Although the "LiveCode Server" [1] is intended primarily as a CGI processor, but you can use it quite easily for command line applications.

This page describes the steps required to do this.

Installation as a command line processor

The basic "installation" of the "LiveCode Server" is already described in the "LiveCode Lesson" " [How do I install LiveCode Server?](#) ":

1. Download the "LiveCode Server" package for the operating system you are using and
2. unzip the package.

After unpacking, you will see that the "LiveCode Server" essentially consists of an executable program ("livecode-server") as well as two directories with associated libraries ("drivers" and "externals").

In principle, the "LiveCode Server" can already be used now (the "LiveCode Lesson" explains how to do this). However, it would be very annoying to use as an interpreter of command line scripts if you had to change to the directory with the "LiveCode Server" every time and write "livecode-server" before the name of the actual script ...

Fortunately, Unix and Linux provide a comfortable way out of this situation using the so-called "shebang" line. But first you should move the "LiveCode Server" into a system directory:

```
sudo mv <LCServer directory> / * / usr / local / bin
```

So move the contents of the unzipped "LiveCode Server" directory /usr/local/bin. Since the target directory belongs to the operating system, you must execute the command as an administrator (therefore sudo at the beginning).

You should also create /usr/local/bin a (shell) script in the directory called lc-server:

```
sudo cat> / usr / bin / env / lc-server << EOF
#! / usr / bin / env bash
livecode-server "$@" | tail -n +2
EOF
```

This script should be executable:

```
sudo chmod + x / usr / bin / env / lc-server
```

Once that's done, you can put the line at the top of every "LiveCode Server" script

```
#! / usr / bin / env lc-server
```

write. This is the aforementioned "shebang" line and must be at the very beginning of the script file.

The example shown in the "LiveCode Lesson" would look like this:

```
#! / usr / bin / env lc-server
<? lc
put "Hello world!" && the date && the time
?>
```


In addition, the script (let's call it - as suggested in the "LiveCode Lesson" - `myscript.lc`) must still be marked as executable:

```
chmod +x myscript.lc
```

Once that's done, you can call your script from the command line like this:

```
./myscript.lc
```

(the characters `./` in front of the actual file name should ensure that the call works regardless of your `$PATH` settings). As a result, you should get an output of the form

```
Hello World! 1/26/14 8:34 AM
```

receive.

The script can now be executed without an explicit reference to the "LiveCode Server". The filename suffix `.lc` is completely irrelevant - only important

1. the "Shebang" line at the beginning of the script (`#!/usr/bin/env lc-server`)
2. the executability of the script file (`chmod +x <Skript-Datei>`)

With this trick you can now create command line scripts in LiveCode!

Have fun with the "LiveCode Server"!

references

[(RunRev Ltd.)

1 **LiveCode | LiveCode Server Guide**

] (see <http://livecode.com/developers/guides/server/>)

The LiveCode server is an interpreter for LiveCode scripts, which is started from the command line (without graphical user interface) and is intended primarily as a CGI processor (in this way web pages can be edited with LiveCode - so you do not have to learn more PHP).