

# Package: chshli (via r-universe)

July 5, 2024

**Type** Package

**Title** Check Shared Libraries

**Version** 0.0.1

**Date** 2020-05-26

**Author** Dirk Eddelbuettel

**Maintainer** Dirk Eddelbuettel <edd@debian.org>

**Description** R packages can be downloaded from a repository for the 'RStudio Package Manager' or 'RSPM' which provides prebuilt binaries for several Linux distributions easing installing considerably. However, these packages do not have system level support for their run-time dependencies. This packages aims to address this shortcoming with a first application for Ubuntu 18.04 aka 'bionic'.

**URL** <http://dirk.eddelbuettel.com/code/chshli.html>

**BugReports** <https://github.com/eddelbuettel/chshli/issues>

**License** GPL (>= 2)

**RoxygenNote** 6.0.1

**Repository** <https://eddelbuettel.r-universe.dev>

**RemoteUrl** <https://github.com/eddelbuettel/chshli>

**RemoteRef** HEAD

**RemoteSha** fa842b4fb4fe91eb35e4b246d4096df9fc60aa40

## Contents

checkSharedLibs . . . . . 2

**Index** . . . . . 3

---

|                 |                               |
|-----------------|-------------------------------|
| checkSharedLibs | <i>Check Shared Libraries</i> |
|-----------------|-------------------------------|

---

**Description**

Check shared libraries on current system

**Usage**

```
checkSharedLibs(vec, db)
```

**Arguments**

|     |  |
|-----|--|
| vec | A (optional) character vector of shared libraries, typically with major soname. If missing a default vector is used as fallback. |
| db  | An optional identifier for a library package database, should be one of "18.04" or "19.10".                                      |

**Details**

This function examines the shared libraries in argument `vec`. It runs `ldd` on these, filters for “not found” to identify unresolved shared libraries and tries to map these against (stored) information on distribution packages (where Ubuntu 18.04 and 19.10 are currently supported).

The internal helper function `.libs()` can be used to identify shared libraries from packages in the current `.libPaths()`. Because search all packages in each `.libPaths()` entry at once (using shell tools) it does not currently associate finds with the packages they originate from.

**Value**

Nothing currently but information is printed.

**Author(s)**

Dirk Eddelbuettel

**Examples**

```
checkSharedLibs(c("libxml2.so.2"))
```

# Index

`checkSharedLibs`, [2](#)