

# Multisystemfunktionalität von R Paketen

---

Clemens Schmid

20. Oktober 2016

## Problemstellung

---

- R Paket recexcavAAR
  - Fragestellung, Funktionen und warum ist das nützlich?
  - Rcpp
  - automatisierte Tests
  - Vignettes
  - **CRAN** (Comprehensive R Archive Network)
    - Vorbereitung
    - Kommunikation
    - **Multisystemfunktionalität**
- Multisystemfunktionalität ist mein jüngstes Problem - am deutlichsten präsent und am wenigsten verstanden.
- These: Viele Fehler sind durch gute Vorbereitung und Checks vermeidbar.

- Ein R Paket muss auf möglichst vielen Zielsystemen lauffähig sein.
- Probleme
  - OS (Mac OS, Windows, Linux, Solaris?!)
  - R Version + R-devel vs. R-release
  - Fremdpakete und Abhängigkeiten
- CRAN Checks
  - CRAN Check Systeme
  - recexcavAAR Check Ergebnisse

## Beispiel 1: Solaris 'overloading ambiguity'

FALSCH:

```
int lres2 = pow(lres, 2);  
int lres3 = pow(lres, 3);
```

*Mathematical functions such as sqrt are defined in C++ for floating-point arguments. It is legitimate in C++ to overload these with versions for types float, double, long double and possibly more. **This means that calling sqrt on an integer type may have 'overloading ambiguity' as it could be promoted to any of the supported floating-point types: this is commonly seen on Solaris, but for pow also seen on OS X. (C++98 has an overload for std::pow(, ), but this may not be visible from the main namespace. C++11 requires additional overloads for integer types, and ambiguous overloads are more common in C++11 (and later) compiler modes.)***

– **CRAN: Writing R Extensions §1.6.4**

RICHTIG: Type conversion

```
int lres2 = pow((double)lres, 2.0);  
int lres3 = pow((double)lres, 3.0);
```

## Beispiel 2: plotly Versionsupdate (Vignette)

**FALSCH:**

```
library(plotly)

vis %>%
  add_trace(
    x = surf[[1]]$x, y = surf[[1]]$y, z = surf[[1]]$z, type = "surface",
    showscale = FALSE, opacity = 0.9, hoverinfo = "none"
  )
```

**Github Issue**

**RICHTIG:**

```
library(rgl)

surface3d(
  surf[[i]]$x, surf[[i]]$y, t(surf[[i]]$z),
  color = c("black", "white"),
  alpha = 0.5,
  add = TRUE
)
```

## Lösungsansätze

---

- Vorbereitung
  - Hadley Wickhams Anleitung
  - Anleitungen des R Development Core Team -> Manuals -> Writing R Extensions
- Workflow
  - testthat
  - roxygen2
  - **devtools** (`devtools::check_cran()`)
  - etc.
- Testumgebungen
  - Travis CI (recexcavAAR: webGUI + .travis.yml)
  - eigene virtuelle Maschinen (siehe virtual box solaris)
  - Winbuilder (`devtools::build_win()`)
  - Rocker?
  - **ISAAC: Web of Czechs?**