



An Investigation of the R Package Ecosystem

Florian Sihler, Oliver Gerstl, Julian Schubert, Thomas Schöller, Matthias Tichy | useR! 2026

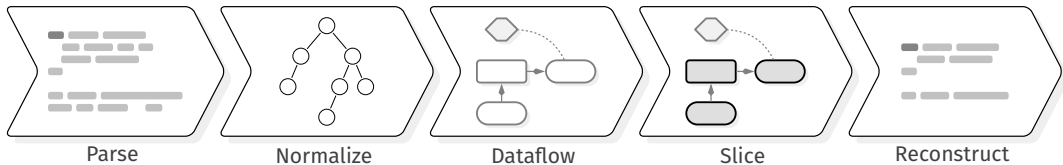


Software Engineering
Programming Languages

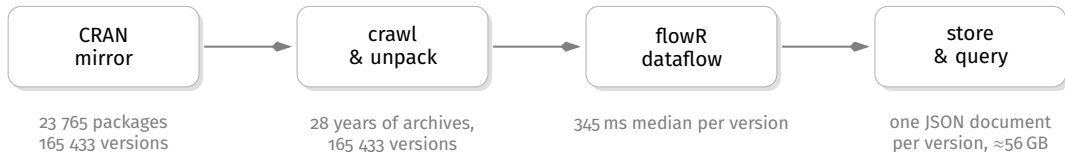


universität
uulm

flowR: Static Analysis for a Dynamic Language



How We Gather the Data



For every version of every package we built a

- **dataflow**,
- **call**, and
- **control-flow graph**.

Crawling 28 Years of CRAN

Full Mirror

including base packages

165 433 Versions

unpacking their sources

15.3 billion Downloads

linked to every version

Inspecting the Graphs



On top of them, 22 inspectors and 74 semantic queries extract, for every single version:

Call Graphs

224 million calls

Dependencies

399 million edges

Language Features

607 tracked

Dead Code

≈100 000 uncalled functions

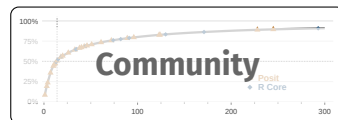
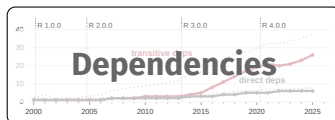
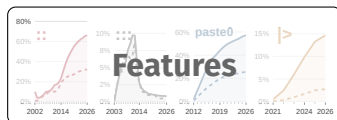
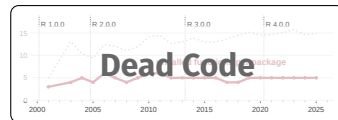
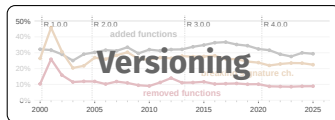
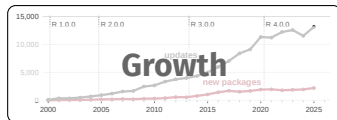
Changelogs

≈115 000 release notes

Package Summaries

23 765 profiled

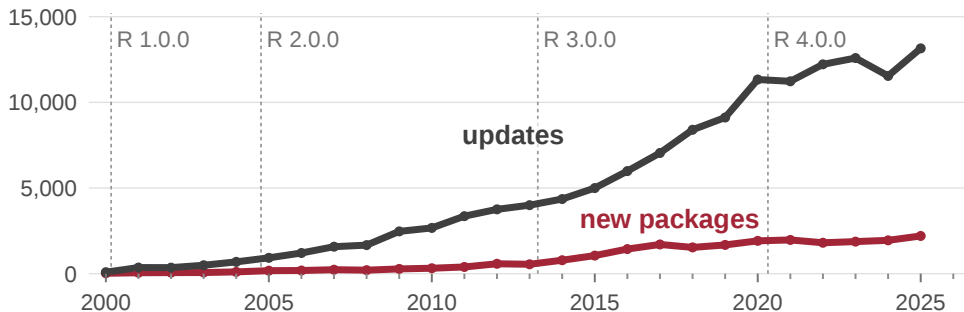
The Ecosystem at a Glance



A maturing, ever-growing ecosystem that is remarkably stable.

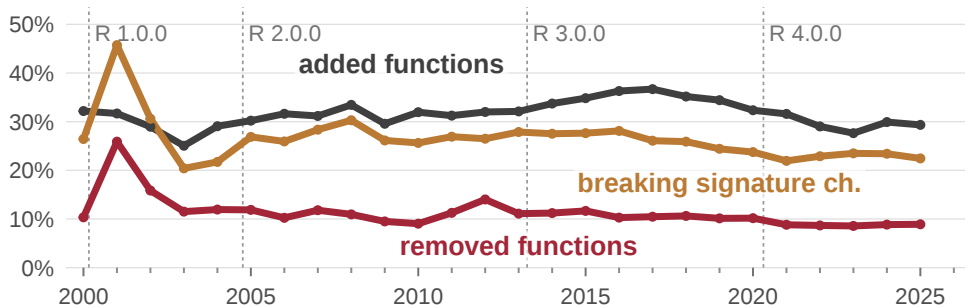


Ecosystem Growth



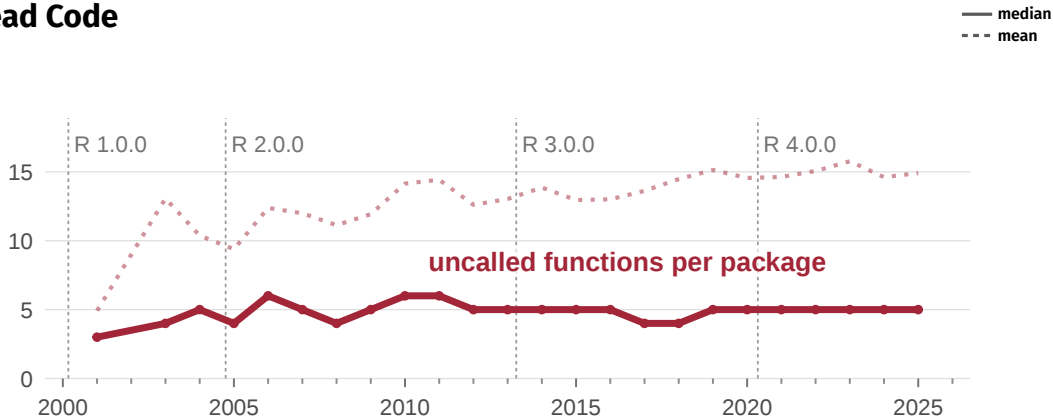
Updates now dominate: only $\approx 24\%$ of 2025 releases are new packages.

(Semantic) Versioning in Practice



29.6 % of consecutive versions break the API, even 23.4 % of patch releases.

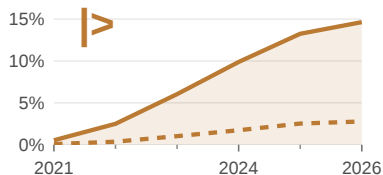
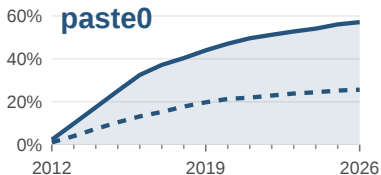
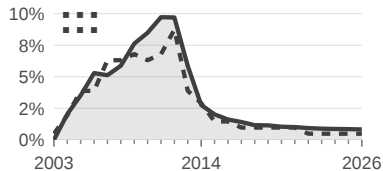
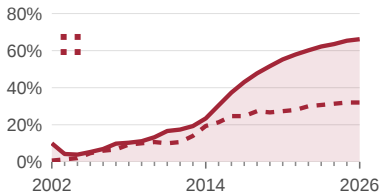
Dead Code



CRAN ships nearly 100 000 functions that are never called.

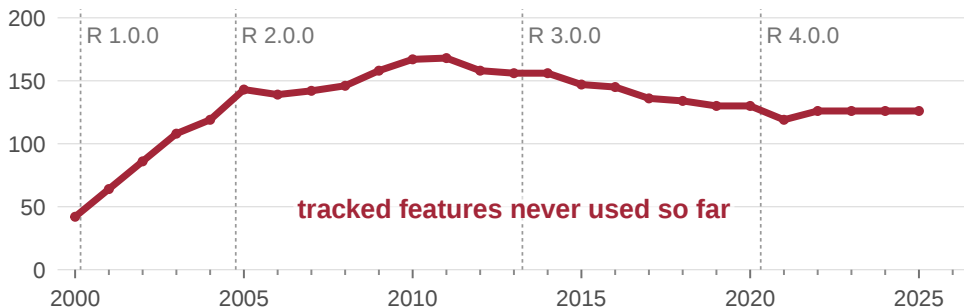
Language Feature Adoption

— new packages
- - - old packages



New packages adopt faster. :: : vanished once CRAN flagged it.

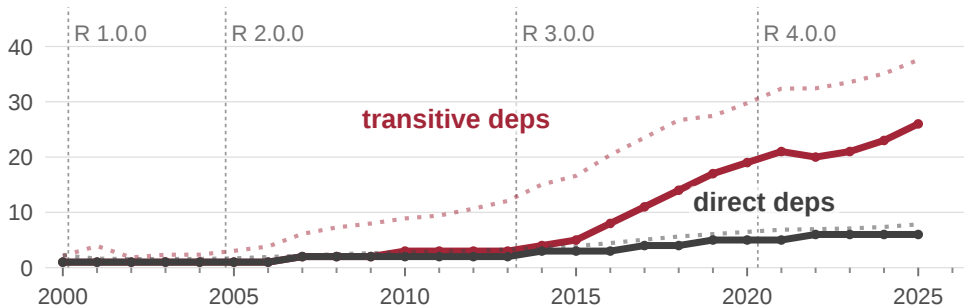
Never-Used Language Features



126 of 607 tracked features remain unused (in production packages).

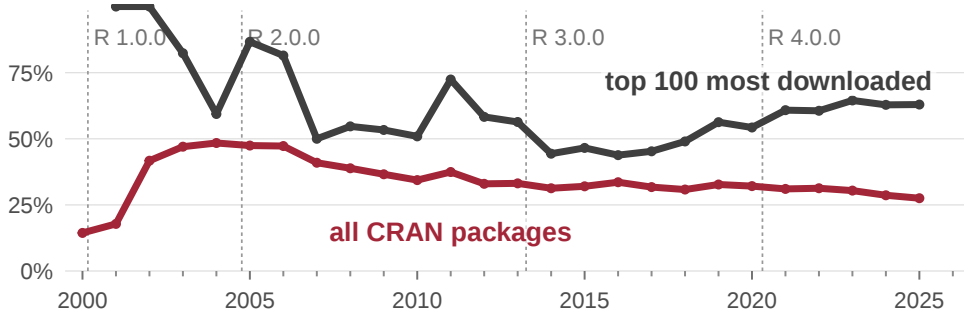
Dependency Growth

— median
- - - mean



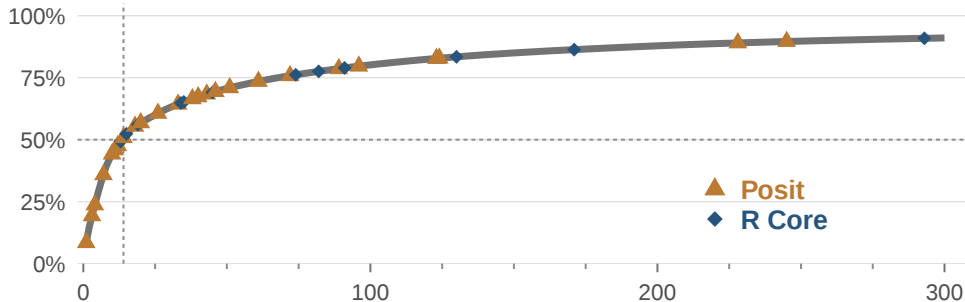
Every upstream breaking change ripples further each year.

Version Locking



**Ecosystem-wide, only 27.5 % of declarations pin a version,
yet the 100 most downloaded lock far more (62.9 %).**

Community & Truck Factor



14 maintainers cover half of all CRAN downloads, just 5 a quarter.

Security

1 in 4

packages calls security-critical
functions like `eval` or `system`

0.1%

of changelogs mention security

Open Challenges for the Ecosystem

Reproducibility

across package versions

Silent Breakage

versions understate change

Dead Code & Dependencies

both keep growing unchecked

Security

surfacing sensitive calls

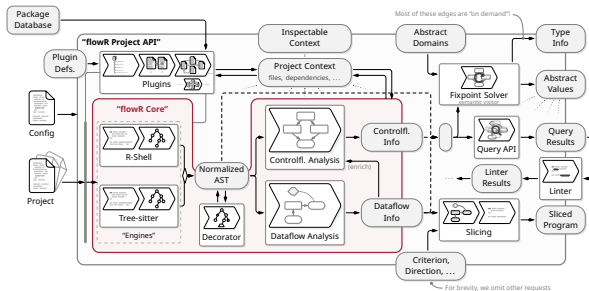
Takeaways

The ecosystem's biggest risks are **semantic**, and invisible on the surface:

- CRAN ships **≈100 000** functions that are never called
- **29.6 %** of consecutive versions silently break the API
- **1 in 4** packages call `eval` or `system`

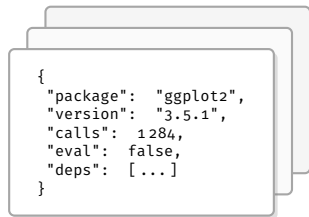
Seeing any of this needs **static analysis** and **transitive call graphs**.

Back to flowR & the Package Database



flowR — an open static-analysis engine for R

github.com/flowr-analysis/flowr



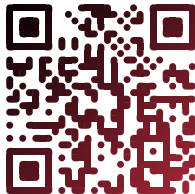
Every version, queryable —

23 765 packages • 165 433 versions • 56 GB

Asking the Community

Many interesting queries remain open.

Which questions matter most to *you*?



An Investigation of the R Package Ecosystem

github.com/flowr-analysis/flowr

