Package: rxeval (via r-universe)

November 17, 2026

Title Evaluate 'ODK'/'OpenRosa' 'XForms' Logic

Version 0.1.3

Description Check what a questionnaire's own rules say about a submission, and which of its expressions will behave differently on a tablet than in a browser. Wraps the 'rxeval' 'Rust' crate.

License BSD_2_clause + file LICENSE

Encoding UTF-8

RoxygenNote 8.0.0

Config/rextendr/version 0.5.0

SystemRequirements Cargo (Rust's package manager), rustc >= 1.65.0, xz

Depends R (>= 4.2)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://milkway.github.io/rxeval/r/, https://github.com/milkway/rxeval, https://milkway.github.io/rxeval/

BugReports https://github.com/milkway/rxeval/issues

NeedsCompilation yes

Author André Leite [aut, cre] (ORCID: <https://orcid.org/0000-0002-4718-9766>), Raydonal Ospina [aut] (ORCID: <https://orcid.org/0000-0002-9884-9090>), Cristiano Ferraz [aut] (ORCID: <https://orcid.org/0000-0002-6838-6734>), The authors of the vendored Rust crates [cph] (see inst/AUTHORS for details)

Maintainer André Leite <leite@castlab.org>

Config/pak/sysreqs xz-utils libclang-dev

Repository https://cran.r-universe.dev

Date/Publication 2026-11-17 12:00:07 UTC

RemoteUrl https://github.com/cran/rxeval

RemoteRef HEAD

RemoteSha 989a6352dfa7ffa050a67fd23de38e77ea87cc08

Contents

eval_expression 2
form_portability 3
rust_eval_expression 3
rust_form_portability 4
rust_submission_findings 4
submission_findings 5

Index **7**

`eval_expression` *Evaluate one XPath expression against a submission*

Description

For working out what a rule does before writing it into a form.

Usage

```
eval_expression(expression, submission, today = NULL, now = NULL)
```

Arguments

`expression`	The XPath expression, in the ODK/OpenRosa dialect.
`submission`	Path to a submission, or its XML.
`today`, `now`	What `today()` and `now()` should answer.

Value

A length-one character vector. An expression this engine cannot evaluate raises an error rather than returning a plausible value.

Examples

```
submission <- system.file("extdata", "visit-submission.xml", package = "rxeval")
eval_expression("count(/data/resident)", submission)
eval_expression("/data/residents > count(/data/resident)", submission)
```

form_portability *Will this form mean the same thing on a tablet and in a browser?*

Description

The two engines the ODK ecosystem runs on implement different languages: JavaRosa inside Collect and KoboCollect, Enketo inside web forms. Neither is a superset of the other, and the gaps do not announce themselves — an expression Collect cannot evaluate usually yields nothing rather than an error, so the form fills in, the interview finishes, and a column comes back empty.

Usage

```
form_portability(form)
```

Arguments

`form`	Path to an XForm, or the XML itself.

Details

Every rule behind this was found by putting the same expression to both engines and reading the two answers.

Value

A data frame with one row per issue. `breaks` says which side stumbles: `"Collect / KoboCollect"`, `"Enketo web forms"`, or `"both, differently"` — the last being the worst, since nothing fails anywhere. `suggestion` is a rewrite that means one thing in both, or `NA` when there is none to give. Zero rows means the form travels.

Examples

```
form <- system.file("extdata", "visit.xml", package = "rxeval")
issues <- form_portability(form)
issues[, c("path", "breaks", "suggestion")]
```

rust_eval_expression *Evaluate one XPath expression against a submission.*

Description

For working out what a rule actually does before writing it into a form.

Usage

```
rust_eval_expression(expression, submission, today, now)
```

Arguments

expression	The XPath expression.
submission	Path to a submission, or its XML.
today, now	What today() and now() should answer.

Value

A length-one character vector. An expression the engine cannot evaluate raises an error rather than returning a plausible value.

---

rust_form_portability *Which expressions in a form will behave differently on a tablet than in a browser.*

Description

Which expressions in a form will behave differently on a tablet than in a browser.

Usage

```
rust_form_portability(form)
```

Arguments

form	Path to an XForm, or the XML itself.

Value

A data frame with one row per issue: path, rule, expression, construct, breaks, effect, suggestion, says. Zero rows means the form travels.

---

rust_submission_findings *What a form's own rules say about a submission.*

Description

What a form's own rules say about a submission.

Usage

```
rust_submission_findings(form, submission, today, now)
```

Arguments

form	Path to an XForm, or the XML itself.
submission	Path to a submission, or its XML.
today, now	What today() and now() should answer. Defaults to the submission's own metadata, so a date rule is judged against the day the work was done rather than the day of the check.

Value

A data frame: path, kind, says. Zero rows means the submission satisfies every rule the engine could evaluate — and a rule it could not evaluate appears as kind not-evaluated, never as a pass.

---

submission_findings *What a form's own rules say about a submission*

Description

Evaluates the questionnaire's relevant, constraint, required and calculate against collected data, and reports what does not hold.

Usage

```
submission_findings(form, submission, today = NULL, now = NULL)
```

Arguments

form	Path to an XForm, or the XML itself.
submission	Path to a submission, or its XML.
today, now	What today() and now() should answer. Left NULL, they come from the submission's own metadata, so a date rule is judged against the day the work was done rather than the day of the check — otherwise a submission that was valid on collection starts failing later, and the report changes while the data does not.

Details

A rule the engine could not evaluate comes back as not-evaluated, never as a pass: a rule that did not run has not been satisfied, and reporting it as clean would be the one failure worth avoiding.

Value

A data frame of path, kind (constraint, required, calculation, not-evaluated) and says. Zero rows means every rule held.

Examples

```
form <- system.file("extdata", "visit.xml", package = "rxeval")
submission <- system.file("extdata", "visit-submission.xml", package = "rxeval")
findings <- submission_findings(form, submission)
findings[, c("path", "kind")]
```

Index

eval_expression, 2

form_portability, 3

rust_eval_expression, 3
rust_form_portability, 4
rust_submission_findings, 4

submission_findings, 5