Package: form.model (via r-universe)

November 11, 2026

Title Neutral Versioned Form Definitions and Workbook Layouts

Version 0.6.3

Description Loads explicit YAML resources describing versioned forms, fields, declarative constraints, reference requirements, and workbook layouts. Checks model definitions against an installed YAML schema and consistency rules, and provides value-like inspection APIs. Does not validate submitted data or connect to external services.

License LGPL-2.1

URL https://ultreia-io.github.io/form.model/, https://github.com/ultreia-io/form.model

BugReports https://github.com/ultreia-io/form.model/issues

Encoding UTF-8

Depends R (>= 4.1.0)

Imports yaml (>= 2.3.10)

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Roxygen list(markdown = TRUE)

Config/testthat/edition 3

Config/Needs/coverage covr, xml2

Config/Needs/website pkgdown

Config/roxygen2/version 8.1.1

Repository https://ultreia-io.r-universe.dev

Date/Publication 2026-11-11 09:37:46 UTC

RemoteUrl https://github.com/ultreia-io/form.model

RemoteRef HEAD

RemoteSha fe6c82d4a5b077066892ac45ee821597d9445e6a

Contents

as_form_model . . . 2
check_form_model . . . 3
form_definition . . . 3
form_field . . . 4
form_fields . . . 5
form_layout . . . 5
form_metadata . . . 6
form_references . . . 7
form_resource . . . 7
form_schema . . . 8
form_sheets . . . 8
form_value_matches_type . . . 9
load_form_model . . . 10
print.form_model . . . 11
print.form_model_check . . . 11
read_form_model . . . 12
select_form_model . . . 12

Index **14**

`as_form_model` *Construct a checked model*

Description

Construct a checked model

Usage

```
as_form_model(definition, resource = "<list>")
```

Arguments

`definition`	Named list following `form_schema()`; sequences are unnamed lists.
`resource`	Source label for diagnostics.

Value

The definition as a `form_model` list, without the source label. Invalid definitions raise `model_form_error` with diagnostics.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Construct a checked model from the definition.
as_form_model(form_definition(model_definition))
```

check_form_model *Check a form definition*

Description

Check a form definition

Usage

```
check_form_model(definition, resource = "<list>")
```

Arguments

definition	Named list following form_schema(); sequences are unnamed lists.
resource	Source label for diagnostics.

Details

Checks schema structure, then model consistency if structure passes. Invalid definitions return diagnostics; invalid arguments raise model_form_error. Submitted data is not checked.

Value

A form_model_check with valid and a diagnostics data frame containing resource, path and message columns.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Check the structure and consistency of the definition.
check_form_model(form_definition(model_definition))
```

form_definition *Get the complete definition*

Description

Get the complete definition

Usage

```
form_definition(model)
```

Arguments

model	A checked form_model from read_form_model() or as_form_model().

Details

Accessors preserve declaration order and do not revalidate models. Treat checked models as read-only. Invalid model arguments raise `model_form_error`.

Value

An ordinary model definition with `form`, `sheets`, and `references` containing named catalogues and their declared tables.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Inspect the declared form identity.
form_definition(model_definition)$form
```

`form_field` *Get one field*

Description

Get one field

Usage

```
form_field(model, id)
```

Arguments

`model`	A checked `form_model` from `read_form_model()` or `as_form_model()`.
`id`	Exact `<sheet>.<field>` identifier.

Value

One definition as returned by `form_fields()`. Unknown IDs raise `model_form_error`.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Inspect the allowed values for the observation unit.
form_field(model_definition, "Observations.unit")$enum
```

form_fields *Get fields in declaration order*

Description

Get fields in declaration order

Usage

```
form_fields(model, sheet = NULL)
```

Arguments

model	A checked form_model from read_form_model() or as_form_model().
sheet	Exact worksheet name, or NULL for all sheets.

Details

Adds field IDs, names, sheet names and numeric coordinates. Omitted mandatory becomes FALSE; reference severity defaults to "error". Use form_definition() for the original declarations.

Value

Field definitions named <sheet>.<field>. Unknown sheets raise model_form_error.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Inspect the observation fields in workbook column order.
form_fields(model_definition, sheet = "Observations")
```

form_layout *Get a sheet layout*

Description

Get a sheet layout

Usage

```
form_layout(model, id)
```

Arguments

model	A checked form_model from read_form_model() or as_form_model().
id	Exact worksheet name.

Value

The effective layout (`cells` or `table`), with table `column_count` derived from the fields. Unknown sheets raise `model_form_error`.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Inspect the observation table layout.
form_layout(model_definition, "Observations")
```

`form_metadata` *Get form identity and versions*

Description

Get form identity and versions

Usage

```
form_metadata(model)
```

Arguments

`model`	A checked `form_model` from `read_form_model()` or `as_form_model()`.

Value

A list with `form` and `schema_version`.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Inspect the model identity and declared reference catalogues.
form_metadata(model_definition)
```

form_references *Get declared reference catalogues and tables*

Description

Get declared reference catalogues and tables

Usage

```
form_references(model)
```

Arguments

model	A checked form_model from read_form_model() or as_form_model().

Value

A named list of catalogue names and their declared table identifiers. Reference data is not loaded.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Inspect the catalogue and table requirements.
form_references(model_definition)
```

form_resource *Locate an installed resource*

Description

Locate an installed resource

Usage

```
form_resource(path, package)
```

Arguments

path	File path relative to the package directory.
package	Installed package name.

Value

Absolute file path. Missing files and paths escaping the package raise model_form_error.

Examples

```
# Locate the installed survey definition.
form_resource("examples/survey/form.yaml", "form.model")
```

`form_schema` *Read the model-definition schema*

Description

Read the model-definition schema

Usage

```
form_schema(version = "1.0.0")
```

Arguments

`version`	Schema version; currently `"1.0.0"`.

Details

Uses the package's `form.model/schema/1` dialect, not JSON Schema.

Value

The schema as a nested list. Unsupported versions raise `model_form_error`.

Examples

```
# Inspect the supported schema contract version.
form_schema()$contract_version
```

`form_sheets` *Get sheet definitions*

Description

Get sheet definitions

Usage

```
form_sheets(model)
```

Arguments

`model`	A checked `form_model` from `read_form_model()` or `as_form_model()`.

Details

Table layouts include `column_count` derived from the ordered fields. An optional `primary_key` preserves the declared sequence of local field names. Key components must be filled and their combination must identify one row; this package checks the declaration, not submitted values. `unique_key` preserves an ordered key whose complete values must be unique. `at_least_one` preserves selected fields and optional severity (default error); populated rows must supply at least one selected value. Use `form_definition()` for the original declarations.

Value

Sheets in declaration order, named by worksheet, each with `fields` and `layout`.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# List the worksheet names in workbook order.
names(form_sheets(model_definition))
```

---

form_value_matches_type

Check a scalar against a form field type

Description

Share field-type checks between model definitions and validation consumers. Requiredness and other field constraints are handled by the caller.

Usage

```
form_value_matches_type(x, type)
```

Arguments

`x`	Scalar value to check, without coercion.
`type`	One of `string`, `email`, `integer`, `number`, `boolean`, `latitude`, `longitude`, `date`, or `datetime`.

Details

Dates require `YYYY-MM-DD`; datetimes require `YYYY-MM-DDTHH:MM:SSZ`. Both must parse and format back to the identical string, using UTC for datetimes. Emails require nonempty local and domain parts, a domain dot, and no whitespace or additional at signs. Mailbox existence is not checked. Numeric values must be finite; integers must have no fractional part. Coordinates are decimal degrees: latitude is in `[-90, 90]` and longitude in `[-180, 180]`, including both endpoints. Numeric text is not coerced.

Value

One logical value; FALSE for missing or incompatible values, unsupported types, or classed values such as factors, Date, and POSIXct.

Examples

```
# Check whether the value matches the email type.
form_value_matches_type("contact@example.org", "email")
# Check whether the value matches the datetime type.
form_value_matches_type("2026-10-18T12:00:00Z", "datetime")
```

---

load_form_model *Load a model by filename prefix*

Description

Load a model by filename prefix

Usage

```
load_form_model(name, directory)
```

Arguments

name	Prefix of <name>-form.yaml, starting with a letter and containing only letters, digits, underscores, dots or hyphens.
directory	Directory containing the file.

Value

A checked form_model; see read_form_model() for errors.

Examples

```
# Choose a temporary directory for the named model example.
model_directory <- tempfile("named-model-")
# Create the temporary model directory.
dir.create(model_directory)
# Copy the survey definition using the named-loader convention.
file.copy(form_resource("examples/survey/form.yaml", "form.model"),
  file.path(model_directory, "survey-form.yaml"))
# Load the definition using its filename prefix.
load_form_model("survey", model_directory)
# Remove the temporary model directory.
unlink(model_directory, recursive = TRUE)
```

print.form_model *Print a model summary*

Description

Print a model summary

Usage

```
## S3 method for class 'form_model'
print(x, ...)
```

Arguments

x	A checked form_model.
...	Unused.

Value

x, invisibly; prints identity, versions, and field and sheet counts.

print.form_model_check *Print definition check results*

Description

Print definition check results

Usage

```
## S3 method for class 'form_model_check'
print(x, ...)
```

Arguments

x	A form_model_check from check_form_model().
...	Unused.

Value

x, invisibly; prints validity and any diagnostics.

read_form_model *Read and check a YAML form model*

Description

Read and check a YAML form model

Usage

```
read_form_model(path, package = NULL)
```

Arguments

path	File path, or package-relative path when package is supplied.
package	Optional installed package name.

Details

Uses YAML 1.1 types: quote versions and codes; use true/false for booleans. Tags, anchors, aliases, merges, directives and document markers are unsupported. The source path is not stored in the model.

Value

A checked form_model list. Invalid YAML or definitions raise model_form_error with source context and available diagnostics.

Examples

```
# Load the bundled survey model.
model_definition <- read_form_model("examples/survey/form.yaml", "form.model")
# Inspect the model identity and declared reference catalogues.
form_metadata(model_definition)
```

select_form_model *Select an exact model version*

Description

Select an exact model version

Usage

```
select_form_model(paths, id, version, package = NULL)
```

Arguments

paths	Nonempty character vector of YAML file paths.
id	Exact form identifier.
version	Exact form version.
package	Optional installed package containing the relative paths.

Value

The matching form_model. All files are checked; invalid files or anything other than one match raise model_form_error.

Examples

```
# Select the survey definition with the exact requested version.
select_form_model("examples/survey/form.yaml", "survey", "2026.2",
                  package = "form.model")
```

Index

as_form_model, 2
as_form_model(), *3–8*

check_form_model, 3
check_form_model(), *11*

form_definition, 3
form_definition(), *5*, *9*
form_field, 4
form_fields, 5
form_fields(), *4*
form_layout, 5
form_metadata, 6
form_references, 7
form_resource, 7
form_schema, 8
form_schema(), *2, 3*
form_sheets, 8
form_value_matches_type, 9

load_form_model, 10

print.form_model, 11
print.form_model_check, 11

read_form_model, 12
read_form_model(), *3–8*, *10*

select_form_model, 12