

# Package ‘CLDedgelister’

August 20, 2026

**Type** Package

**Title** Import System-Dynamics Models and Convert Them to Edge Lists

**Version** 1.0.3

**Description** Imports causal and system-dynamics models from 'Vensim', 'Stella' / 'iThink' and 'Powersim Studio' ('XMILE'), 'AnyLogic' and 'GoldSim' (exported 'XML'), and converts them into a two-column edge list of cause-to-effect links. 'Vensim' models are read from their native diagram files; 'Powersim Studio' and 'GoldSim' are also now read from their native diagram files in addition to exported 'XMILE' or 'XML'. Provides an 'RStudio' add-in with a simple point-and-click interface, together with command-line functions that return a data frame or write it to 'CSV' or 'Excel'.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Imports** xml2, tools, utils, stats

**Suggests** shiny, miniUI, rstudioapi, writexl, DT, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Config/roxygen2/version** 8.1.0

**NeedsCompilation** no

**Author** Taddias Prince Mpofu [aut, cre, cph],  
Mawuko Bonoukpoe Joel Sokame [aut],  
Frank Chidawanyika [aut],  
Joel Cornelius Runyu [aut],  
Marc Marin Agbodjan [aut]

**Maintainer** Taddias Prince Mpofu <princetaddy@gmail.com>

**Repository** CRAN

**Date/Publication** 2026-08-20 02:10:02 UTC

## Contents

|                              |   |
|------------------------------|---|
| cld_edgelist_addin . . . . . | 2 |
| cld_formats . . . . .        | 2 |

|                                     |           |
|-------------------------------------|-----------|
| looks_like_goldsim_binary . . . . . | 3         |
| looks_like_powersim . . . . .       | 3         |
| looks_like_stm . . . . .            | 4         |
| model_to_edgelist . . . . .         | 5         |
| model_to_edgelist_file . . . . .    | 5         |
| parse_alpx . . . . .                | 6         |
| parse_powersim . . . . .            | 7         |
| parse_stm . . . . .                 | 7         |
| parse_vmfX . . . . .                | 8         |
| powersim_links . . . . .            | 9         |
| powersim_variables . . . . .        | 10        |
| <b>Index</b>                        | <b>11</b> |

---

|                    |                                                                      |
|--------------------|----------------------------------------------------------------------|
| cld_edgelist_addin | <i>Import a model and convert it to an edge list (RStudio addin)</i> |
|--------------------|----------------------------------------------------------------------|

---

**Description**

Opens a small window to pick a Vensim, Stella, Powersim, AnyLogic or GoldSim model, preview the resulting INPUT -> OUTPUT edge list, and optionally save it as CSV or Excel. The chosen edge list is returned as a data.frame when the gadget closes, so it can be captured with `el <- cld_edgelist_addin()`.

**Usage**

`cld_edgelist_addin()`

**Value**

A data.frame of INPUT/OUTPUT edges (invisibly NULL if cancelled). When launched from the RStudio Addins menu the result is returned to the console.

---

|             |                                |
|-------------|--------------------------------|
| cld_formats | <i>Supported model formats</i> |
|-------------|--------------------------------|

---

**Description**

Supported model formats

**Usage**

`cld_formats()`

**Value**

A character vector of the format labels the package understands.

**Examples**

```
cld_formats()
```

---

```
looks_like_goldsim_binary
```

*Detect GoldSim's native binary model*

---

**Description**

Detect GoldSim's native binary model.

**Usage**

```
looks_like_goldsim_binary(path)
```

**Arguments**

path                      Path to the model file.

**Details**

Formats are told apart by CONTENT rather than by extension. Powersim and GoldSim both use .sim for different formats: Powersim writes an OLE2 compound document and GoldSim does not, so each file reaches the right reader.

**Value**

A single logical value.

**See Also**

[model\\_to\\_edgelist](#)

---

```
looks_like_powersim
```

*Detect a Powersim Studio project file*

---

**Description**

Detect a Powersim Studio project file. A .sip/.sim is an OLE2 compound document holding IFF-chunked streams, not XMILE. Variable names decode cleanly from the FTXt chunks; the links do not, because Powersim indirects their endpoints through an undocumented symbol table. Export to XMILE from Powersim Studio for a model that can be read exactly.

**Usage**

```
looks_like_powersim(path)
```

**Arguments**

path                      Path to a .sip or .sim file.

**Value**

A single logical value.

---

|                |                                            |
|----------------|--------------------------------------------|
| looks_like_stm | <i>Detect a legacy Stella binary model</i> |
|----------------|--------------------------------------------|

---

**Description**

Before XMILE, isee systems saved models in a proprietary binary format whose files begin with the signature TL2STM2. Such a file is not XML and cannot be read by an XML parser, so it must be routed to [parse\\_stm](#) rather than to the XMILE reader.

**Usage**

```
looks_like_stm(path)
```

**Arguments**

path                      Path to a candidate model file.

**Value**

A single logical value: TRUE when the file carries the legacy Stella binary signature TL2STM2, otherwise FALSE.

**See Also**

[parse\\_stm](#)

**Examples**

```
p <- tempfile(fileext = ".stm")
writeLines("<xmile><model/></xmile>", p)
looks_like_stm(p)
```

---

|                   |                                                             |
|-------------------|-------------------------------------------------------------|
| model_to_edgelist | <i>Convert a system-dynamics model file to an edge list</i> |
|-------------------|-------------------------------------------------------------|

---

**Description**

Reads a model from Vensim, Stella / iThink, Powersim Studio, AnyLogic or GoldSim and returns a two-column data.frame of INPUT (cause) and OUTPUT (effect) links - the same edge list produced by the SDM Network Analyzer.

**Usage**

```
model_to_edgelist(path, format = NULL)
```

**Arguments**

|        |                                                                                        |
|--------|----------------------------------------------------------------------------------------|
| path   | Path to the model file.                                                                |
| format | One of cld_formats(). If NULL (default) the format is guessed from the file extension. |

**Value**

A data.frame with columns INPUT and OUTPUT. Node diagram positions, when available, are attached as attr(x, "positions").

**Examples**

```
# Convert the bundled example Vensim model to an edge list
mdl <- system.file("extdata", "example_model.mdl", package = "CLDedgelist")
el <- model_to_edgelist(mdl)
head(el)
```

---

|                        |                                                             |
|------------------------|-------------------------------------------------------------|
| model_to_edgelist_file | <i>Convert a model file and write the edge list to disk</i> |
|------------------------|-------------------------------------------------------------|

---

**Description**

Convert a model file and write the edge list to disk

**Usage**

```
model_to_edgelist_file(path, out_path = NULL, format = NULL)
```

**Arguments**

|          |                                                                                              |
|----------|----------------------------------------------------------------------------------------------|
| path     | Path to the model file.                                                                      |
| out_path | Output file (.csv or .xlsx). If NULL, the model's name with an _edgelist.csv suffix is used. |
| format   | See <a href="#">model_to_edgelist</a> .                                                      |

**Value**

(invisibly) the edge list data.frame.

**Examples**

```
mdl <- system.file("extdata", "example_model.mdl", package = "CLDedgelist")
out <- tempfile(fileext = ".csv")
model_to_edgelist_file(mdl, out)
```

---

|            |                                                   |
|------------|---------------------------------------------------|
| parse_alpx | <i>Read a compressed AnyLogic project (.alpx)</i> |
|------------|---------------------------------------------------|

---

**Description**

Read a compressed AnyLogic project (.alpx).

**Usage**

```
parse_alpx(path)
```

**Arguments**

|      |                         |
|------|-------------------------|
| path | Path to the model file. |
|------|-------------------------|

**Details**

Formats are told apart by CONTENT rather than by extension. Powersim and GoldSim both use .sim for different formats: Powersim writes an OLE2 compound document and GoldSim does not, so each file reaches the right reader.

**Value**

A two-column data.frame of INPUT and OUTPUT.

**See Also**

[model\\_to\\_edgelist](#)

---

|                |                                                     |
|----------------|-----------------------------------------------------|
| parse_powersim | <i>Read a Powersim Studio project (.sip / .sim)</i> |
|----------------|-----------------------------------------------------|

---

**Description**

Read a Powersim Studio project (.sip / .sim). A .sip/.sim is an OLE2 compound document holding IFF-chunked streams, not XMILE. Variable names decode cleanly from the FTXT chunks; the links do not, because Powersim indirects their endpoints through an undocumented symbol table. Export to XMILE from Powersim Studio for a model that can be read exactly.

**Usage**

```
parse_powersim(path)
```

**Arguments**

|      |                              |
|------|------------------------------|
| path | Path to a .sip or .sim file. |
|------|------------------------------|

**Value**

A two-column data.frame of INPUT and OUTPUT, empty when no links could be recovered, with the variables in the variables attribute.

---

|           |                                                        |
|-----------|--------------------------------------------------------|
| parse_stm | <i>Read a legacy Stella binary model (.STM / .ITM)</i> |
|-----------|--------------------------------------------------------|

---

**Description**

Reads the pre-XMILE binary format used by Stella and iThink, whose files begin with the signature TL2STM2.

**Usage**

```
parse_stm(path)
```

**Arguments**

|      |                              |
|------|------------------------------|
| path | Path to a .STM or .ITM file. |
|------|------------------------------|

## Details

Variable names and equations both come out of the binary cleanly. The mapping BETWEEN them does not: the name records and the equation records are not interleaved one-for-one, and the remaining links are held in an undocumented binary dependency section. Measured on two published models, only 22% and 42% of variables could be matched to their own equation.

Pairing them by proximity anyway yields a network that looks plausible and is partly wrong, with links attributed to the wrong variable. This reader therefore returns an edge list only when the pairing is unambiguous for the great majority of variables, and otherwise stops with an explanation.

To convert such a model, open it in Stella or iThink and use File > Save As to write a .stmxf (XMILE) file, which is read exactly.

## Value

A two-column data.frame with columns INPUT (cause) and OUTPUT (effect), one row per link.

## See Also

[looks\\_like\\_stm](#), [model\\_to\\_edgelist](#)

## Examples

```
p <- tempfile(fileext = ".STM")
writeBin(c(charToRaw("TL2STM2"), as.raw(rep(0, 60)))), p)
looks_like_stm(p)

try(parse_stm(p))
```

---

parse\_vmfxf

*Read a Vensim XML model (.vmfxf)*

---

## Description

Read a Vensim XML model (.vmfxf).

## Usage

```
parse_vmfxf(path)
```

## Arguments

path                      Path to the model file.

## Details

Formats are told apart by CONTENT rather than by extension. Powersim and GoldSim both use .sim for different formats: Powersim writes an OLE2 compound document and GoldSim does not, so each file reaches the right reader.

**Value**

A two-column data.frame of INPUT and OUTPUT.

**See Also**

[model\\_to\\_edgelist](#)

---

|                |                                                      |
|----------------|------------------------------------------------------|
| powersim_links | <i>Read the links from a Powersim Studio diagram</i> |
|----------------|------------------------------------------------------|

---

**Description**

Read the links from a Powersim Studio diagram.

**Usage**

```
powersim_links(path)
```

**Arguments**

|      |                         |
|------|-------------------------|
| path | Path to the model file. |
|------|-------------------------|

**Details**

Formats are told apart by CONTENT rather than by extension. Powersim and GoldSim both use .sim for different formats: Powersim writes an OLE2 compound document and GoldSim does not, so each file reaches the right reader.

**Value**

A two-column data.frame of INPUT and OUTPUT, empty when the file carries no readable diagram.

**See Also**

[model\\_to\\_edgelist](#)

---

|                    |                                                           |
|--------------------|-----------------------------------------------------------|
| powersim_variables | <i>Read variable names from a Powersim Studio project</i> |
|--------------------|-----------------------------------------------------------|

---

**Description**

Read variable names from a Powersim Studio project. A .sip/.sim is an OLE2 compound document holding IFF-chunked streams, not XMILE. Variable names decode cleanly from the FTXT chunks; the links do not, because Powersim indirections their endpoints through an undocumented symbol table. Export to XMILE from Powersim Studio for a model that can be read exactly.

**Usage**

```
powersim_variables(path)
```

**Arguments**

|      |                              |
|------|------------------------------|
| path | Path to a .sip or .sim file. |
|------|------------------------------|

**Value**

A character vector of variable names.

# Index

`cld_edgelist_addin`, [2](#)  
`cld_formats`, [2](#)  
  
`looks_like_goldsim_binary`, [3](#)  
`looks_like_powersim`, [3](#)  
`looks_like_stm`, [4](#), [8](#)  
  
`model_to_edgelist`, [3](#), [5](#), [6](#), [8](#), [9](#)  
`model_to_edgelist_file`, [5](#)  
  
`parse_alpx`, [6](#)  
`parse_powersim`, [7](#)  
`parse_stm`, [4](#), [7](#)  
`parse_vmfx`, [8](#)  
`powersim_links`, [9](#)  
`powersim_variables`, [10](#)