

The **integramme** package

Typesetting logic-grid puzzles in $\text{\LaTeX}$

Jonathan LAFARGE

Version v0.9.0 – 2026-08-23

Abstract

integramme typesets staircase-style logic-grid puzzles, including Einstein-style puzzles. The package separates the logical data from the drawing: authors declare categories, items, clues and, optionally, a solution; the grid is generated automatically. Printed labels may contain ordinary $\text{\LaTeX}$ material while stable internal identifiers are used for references.

1 Status of version 0.9.0

Version 0.9.0 is the first release candidate. The public long-form API and the ordinary rendering defaults are frozen for the planned 1.0 release except for serious defects. Compatibility syntax from earlier prototypes is retained but should not be used in new documents.

Jonathan LAFARGE is the author, copyright holder and CTAN maintainer. The work keeps LPPL status **author-maintained**. Public repository, bug tracker and support URLs are optional CTAN communication metadata and are not invented for this release candidate; they may be added later.

2 Installation

During development, place **integramme.sty** where $\text{\LaTeX}$ can find it. To generate the style file from the documented source, run:

```
latex integramme.ins
```

For a local installation, move the resulting **integramme.sty** to an appropriate directory in your local TEXMF tree and refresh the filename database if your $\text{\TeX}$ distribution requires it.

The runtime dependency is only TikZ. The package requires a $\text{\LaTeX}$ 2e format dated 2022-06-01 or later.

3 Quick start

A minimal blank puzzle is declared by listing categories. Every category currently has to contain the same number of items.

```

\begin{integramme}
  \IntegrammeCategory{person}{Person}{
    alice={Alice},
    bruno={Bruno},
    clara={Clara}
  }
  \IntegrammeCategory{drink}{Drink}{
    tea={Tea},
    coffee={Coffee},
    juice={Juice}
  }
  \IntegrammeCategory{pet}{Pet}{
    cat={Cat},
    dog={Dog},
    rabbit={Rabbit}
  }
\end{integramme}

```

The result is generated from the data; no TikZ coordinates are needed.

		Pet			Drink		
		Cat	Dog	Rabbit	Tea	Coffee	Juice
Person	Alice						
	Bruno						
	Clara						
Drink	Tea						
	Coffee						
	Juice						

4 Einstein's riddle

The well-known “Einstein riddle”, also called the five-houses puzzle, is a natural stress test for a logic-grid package. Its attribution to Albert Einstein is not established; the variant used here follows the data described by the French Wikipedia article “Énigme d’Einstein”, consulted on 2026-08-23. The clue wording below is independently paraphrased rather than copied from that page.

Reference: [French Wikipedia – Énigme d’Einstein](#).

Since version 0.8.1, the package includes this classic puzzle as a convenience preset. The preset only declares ordinary categories, clues and solution rows; it does not introduce a different drawing engine or solving algorithm.

```

\begin{integramme}[
  preset=einstein,
  preset-language=fr

```

```
]
\end{integramme}
```

The same preset can be rendered in English with `preset-language=en`. The default preset language is English. Passing `mode=solution` prints the complete solved grid.

The full puzzle contains six categories of five items, including house position, so it is substantially larger than the quick-start example. The following compact settings keep the demonstration inside the manual:

		Animal	Cigare	Boisson	Couleur	Maison
		Chiens Oiseaux Chats Cheriel Poissonrouge Pall Mall Dunhill Blend Bluemaster Prince		Thé Café Lait Bière Eau	Rouge Verte Blanche Jaune Bleue	Maison 1 Maison 2 Maison 3 Maison 4 Maison 5
Nationalité	Britannique					
	Suédois					
	Danois					
	Norvégien					
	Allemand					
Maison	Maison 1					
	Maison 2					
	Maison 3					
	Maison 4					
	Maison 5					
Couleur	Rouge					
	Verte					
	Blanche					
	Jaune					
	Bleue					
Boisson	Thé					
	Café					
	Lait					
	Bière					
	Eau					
Cigare	Pall Mall					
	Dunhill					
	Blend					
	Bluemaster					
	Prince					

1. La maison rouge est celle du Britannique.
2. Le Suédois possède des chiens.
3. Le Danois boit du thé.
4. La maison verte se trouve immédiatement à gauche de la maison blanche.
5. Dans la maison verte, on boit du café.
6. La personne qui fume des Pall Mall possède des oiseaux.
7. Les Dunhill sont fumées dans la maison jaune.
8. Du lait est bu dans la troisième maison.
9. Le Norvégien occupe la première maison, à l'extrémité gauche.
10. Le fumeur de Blend habite à côté de la personne qui possède des chats.
11. La personne qui possède un cheval vit à côté du fumeur de Dunhill.
12. Le fumeur de Bluemaster boit de la bière.
13. L'Allemand fume des Prince.
14. La maison du Norvégien est voisine de la maison bleue.
15. Le fumeur de Blend a pour voisin quelqu'un qui boit de l'eau.

Question : Qui possède le poisson rouge ?

In this variant the German owns the fish. A standalone, more generously spaced version is shipped as `example-einstein.tex`.

5 Categories and items

`\IntegrammeCategory` `\IntegrammeCategory{<category id>}{<printed title>}{<items>}`

New documents should give every item an internal identifier and a printed label:

```
\IntegrammeCategory{city}{City}{
  st_etienne={Saint-Étienne},
  calais={Calais},
  frejus={Fréjus}
}
```

Internal identifiers are used by manual marks and solutions. Printed labels are independent and may contain commas, slashes, mathematics and ordinary formatting commands. For maximum robustness, keep identifiers short and ASCII-like and avoid commas and equals signs in identifiers.

The older shorthand

```
\IntegrammeCategory{city}{City}{Calais,Fréjus,Annecy}
```

remains accepted for compatibility; in that form the identifier and printed label are the same token list.

6 Clues

`\IntegrammeIndices` `\IntegrammeIndices{<item list>}`

Clues are written as ordinary `\item` entries. They are printed as a numbered list below the grid.

```
\IntegrammeIndices{%
  \item Alice does not drink tea.
  \item The owner of the rabbit drinks coffee.
  \item Clara is not the owner of the dog.
}
```

Because the argument contains normal list items, clue text may contain commas, semicolons and mathematics without introducing a package-level separator.

7 Modes

The key `mode` selects what is printed in the cells.

mode=blank

Prints an empty pupil grid. This is the default.

mode>manual

Prints only marks explicitly declared with `\IntegrammeYes` and `\IntegrammeNo`.

mode=solution

Prints a complete positive/negative grid derived from declared solution rows.

7.1 Manual marks

`\IntegrammeYes` `\IntegrammeYes{<category=item>}{<category=item>}`
`\IntegrammeNo` `\IntegrammeNo{<category=item>}{<category=item>}`

Example:

```
\begin{integramme}[mode>manual]
... categories ...
\IntegrammeNo {person=alice}{drink=tea}
\IntegrammeYes{person=bruno}{pet=dog}
\end{integramme}
```

The two references may be supplied in either order. The package reports an error if a reference is unknown, compares two items from the same category, or declares the same cell both positive and negative.

7.2 Complete solutions

<code>\IntegrammeSolutionRow</code>	<code>\IntegrammeSolutionRow{<category=item,...>}</code>
-------------------------------------	--

The recommended solution syntax is keyed and order-independent:

```
\begin{integramme}[mode=solution]
... categories ...
\IntegrammeSolutionRow{
  person=alice, drink=tea, pet=cat
}
\IntegrammeSolutionRow{
  pet=dog, person=bruno, drink=coffee
}
\IntegrammeSolutionRow{
  drink=juice, pet=rabbit, person=clara
}
\end{integramme}
```

Every solution row must assign exactly one item from every category. A complete solution must contain exactly as many rows as there are items in each category, and every item must occur exactly once per category.

8 Configuration

Package settings can be set globally with `\integrammeset` and overridden locally in the optional argument of the environment.

```
\integrammeset{
  cell-size=6mm,
  column-labels=auto,
  top-angle=45
}

\begin{integramme}[cell-size=5mm]
...
\end{integramme}
```

8.1 Geometry and rules

Key	Default	Meaning
cell-size	6mm	Width and height of one logical cell.
rule-width	0.3pt	Width of thin internal grid lines.
block-rule-width	1pt	Width of category/block borders.
side-title-width	7mm	Width reserved for vertical category titles.
side-label-width	0pt	Fixed row-label width; 0pt requests automatic measurement.
side-label-min-width	26mm	Minimum automatic row-label width.
side-padding	1.5mm	Horizontal padding inside the row-label area.
top-padding	1mm	Padding used in the top-label area.
top-label-min-height	7mm	Minimum height of the top item-label area.
title-height	7mm	Height of the top category-title row.
indices-sep	3mm	Space between the grid and the clue list.
guide-stem-height	0pt	Vertical stem before a slanted guide; 0pt means automatic.

8.2 Header behaviour

Key	Default	Meaning
column-labels	auto	auto, horizontal or slanted. In auto, all top labels become slanted if at least one does not fit horizontally.
top-angle	45	Angle, in degrees, used for slanted top labels.
guides	true	Shows or hides the top label guides.
guide-length	1.15	Slanted guide length, measured in multiples of cell-size.
guide-opacity	0.45	Opacity of the top guides.

8.3 Typography, colours and marks

Key	Default	Meaning
font	\sffamily	Font commands used for ordinary item labels.
title-font	\sffamily\bfseries	Font commands used for category titles.
indices-font	\normalfont	Font commands used for the clue list.
mark-font	\sffamily	Font commands used for positive/negative marks.
yes-mark	\bullet	Material printed for a positive cell (math-safe by default).
no-mark	\times	Material printed for a negative cell (math-safe by default).
grid-color	black	TikZ colour used for the grid.
text-color	black	TikZ colour used for labels and marks.

8.4 Behaviour

Key	Default	Meaning
<code>mode</code>	<code>blank</code>	<code>blank</code> , <code>manual</code> or <code>solution</code> ; selects cell content.
<code>preset</code>	<code>none</code>	<code>none</code> or <code>einstein</code> ; preloads a complete built-in puzzle data set.
<code>preset-language</code>	<code>en</code>	<code>en</code> or <code>fr</code> ; language used by built-in preset labels and clues.
<code>overflow-warning</code>	<code>true</code>	Warns when the complete puzzle is wider than the current <code>\linewidth</code> .

The package never scales a puzzle silently. If the width warning fires, authors remain in control of cell size, label width and surrounding layout.

9 Compatibility syntax

The following forms are still accepted but are not recommended for new documents:

```
\IntegrammeCategory{person}{Person}{Alice,Bruno,Clara}  
\IntegrammeNo{person/Alice}{pet/Cat}  
\IntegrammeSolutionRow{Alice,Tea,Cat}  
\IntegrammeSolution{Alice/Tea/Cat;Bruno/Coffee/Dog;Clara/Juice/Rabbit}
```

Short aliases beginning with `\Int` also remain available for compatibility and convenience, but the long-form commands are the candidate stable API.

10 Diagnostics and current limitations

Version 0.7.0 and later validate the data before drawing. Among other checks, it reports duplicate category or item identifiers, unequal category sizes, malformed or unknown references, conflicting manual marks, incomplete solution rows and duplicate use of solution items. Category identifiers are trimmed before storage, an empty category identifier is rejected explicitly, and a non-positive `cell-size` is rejected before the layout code can divide by it.

Current limitations are:

- all categories must have the same number of items;
- only the staircase layout is implemented;
- long slanted top labels are not automatically constrained to their own category block;
- identifiers should remain simple and should avoid commas and equals signs.

11 Regression testing

Version 0.9.0 runs the release-candidate regression suite on pdfTeX, LuaTeX and XeTeX. The suite checks the candidate public surface and defaults in addition to canonical and compatibility syntax, global/local option scoping, rich labels, representative puzzle sizes, warnings and every package-defined error path. Diagnostic tests redirect package errors to warnings so that complete transcripts can be compared by `l3build` without stopping the test run.

12 Building the package

The source tree now follows the conventional DocStrip/`l3build` pattern. Useful commands are:

```
latex integramme.ins      # generate integramme.sty
l3build unpack            # generate installable files in the build tree
l3build doc               # build integramme.pdf
l3build check             # run regression tests
l3build ctan              # build a CTAN-style archive
```

The release archive produced by `l3build ctan` is intended as the basis for a future CTAN submission. Version 0.9.0 also ships a maintainer-side static audit in the development tree; it checks version consistency, the documented key/command surface and archive contents. Maintainer metadata is finalized; optional repository and support URLs are omitted rather than invented.

13 Release metadata

CTAN requires a public maintainer name for a submission. Jonathan LAFARGE is the public author, copyright holder and CTAN maintainer. This work keeps LPPL status **author-maintained**, which reserves maintenance to the copyright holder. Repository, bug-tracker and support URLs are optional CTAN communication metadata and are intentionally omitted rather than invented. No end-user command depends on these fields.

14 Authorship and AI-assisted development

The author and copyright holder of `integramme` is Jonathan LAFARGE. OpenAI’s ChatGPT was used as an interactive development assistant for code drafting, refactoring, test generation and the drafting/revision of documentation. The model used for the current development cycle was **GPT-5.6 Sol** (August 2026). Design decisions, review, testing and release responsibility remain with the human author.

ChatGPT is therefore acknowledged as a development tool, not listed as an author, copyright holder or CTAN maintainer.

15 License

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, version 1.3c or later. The latest version of this license is available from <https://www.latex-project.org/lppl/>. The work has LPPL maintenance status `author-maintained`.