

TranslatePilot (EN)

AI translations for Joomla articles — with sample, approval, cost gate and mirroring of menus and modules · Version 0.14.0 · Stand 12.09.2026

- [TranslatePilot at a glance](#)
- [Requirements and installation](#)
- [AI providers and access](#)
- [Setting up multilingualism](#)
- [The translation profile](#)
- [Analysis, sample and real run](#)
- [Mirroring menus, modules, footer and templates](#)
- [Translating automatically \(scheduled task\)](#)
- [Frequently asked questions](#)
- [Changelog](#)

TranslatePilot at a glance

TranslatePilot translates the articles of your Joomla website into further languages using an AI service — and takes care of everything else a multilingual website needs: language associations, menu twins, mirrored modules and, if you use YOOtheme Pro, per-language footers and article templates.

At its core is a simple promise: **nothing expensive and nothing irreversible happens without you having seen and approved it first.**

How TranslatePilot works

- **A wizard guides.** The translation wizard leads through everything in eight steps — access, languages, setup, profile, scope, sample, mirroring and the run. Every step judges its own state; only when the required settings stand does it move on.
- **Analysis as a report.** A dedicated view shows what a run would do: which articles are new, which have changed since their last translation, what gets skipped and what the run is expected to cost. The analysis only reads — it writes nothing and spends nothing.
- **Sample before the real run.** Before anything is written for the first time, you translate a few articles as a sample. Source and translation appear side by side; only when the tone fits do you approve the translation profile.
- **Continuation instead of a one-off.** TranslatePilot records per article which state was translated. When an article changes, only it gets retranslated — manually or automatically through the task scheduler.
- **Several safeguards.** A cost limit per run, an upper bound of translated articles per run, a backup confirmation before every real run and the profile approval. Each of them can stop a run before anything happens.

What gets translated — and what does not

Translated are articles of the source language: title, intro and full text (for YOOtheme builder pages the text pieces of the page tree, the structure stays untouched), meta description and keywords, image alternative texts, the labels of the related links, and custom fields of the types text, textarea and editor. Tags, images, link addresses and visibility are taken over unchanged; the language association with the source is created automatically.

Not translated are contents of other components, system texts and configuration. Modules and menu items are not created while translating but through the mirroring steps — as twins of the already translated content. The analysis page shows this boundary before every run.

What TranslatePilot consists of

- **The component** — dashboard with readiness checks, analysis, sample/approval/real run and the mirroring steps.
- **The task plugin** — translates new and changed articles on a schedule through Joomla's task scheduler, with the same safeguards as the backend.

Which AI services are supported

TranslatePilot is provider-open: Claude (Anthropic), Mittwald AI Hosting or any other OpenAI-compatible service. The available models are queried directly from the provider and offered for selection in the options. Details on the page "AI providers and access".

YOOtheme Pro and JSitemap Pro are **optional**. TranslatePilot works completely with the Joomla core; when those extensions are present, they are covered additionally (builder pages, meta overrides, footer and template mirroring).

[Deutsche Fassung](#)

Requirements and installation

Requirements

- Joomla 6.0 or newer
- PHP 8.3 or newer
- Access to a supported AI service (API key) — see "AI providers and access"
- At least two installed Joomla language packs (the source language and every target language)

Your website does **not** have to be configured multilingual to start. TranslatePilot sets up multilingualism itself on request — content languages, language filter, one home page per language and the language switcher. Only the language packs themselves are installed beforehand via *System → Install Languages*; a download cannot honestly be shown in a preview, so the setup does not take it over.

Installation

1. Download the package `pkg_translatepilot-x.y.z.zip`.
2. Install it under *System → Install → Extensions*.
3. The package installs the component and the task plugin; the plugin gets enabled in the process. It stays inactive until you create a scheduled task.

After installation you find TranslatePilot under *Components → TranslatePilot*. The dashboard immediately shows the multilingual state of the installation and what is still missing before the first translation.

Updating

Updates appear as usual under *System → Update → Extensions*. The translation state, the approval and all settings survive updates.

Uninstalling

Uninstalling the package removes the component and the plugin. Already translated articles, menu items and modules are regular Joomla content and naturally remain in place.

[Deutsche Fassung](#)

AI providers and access

TranslatePilot translates through the AI service of your choice. Provider, access and model are configured in the component options (*Components* → *TranslatePilot* → *Options*, tab "Translation").

The providers

- **Claude (Anthropic).** You create the API key in the Anthropic console (`platform.claude.com`). Billing is per token; TranslatePilot shows a USD cost estimate before every run and enforces the cost limit.
- **Mittwald AI Hosting.** You create the API key in mStudio, section AI hosting. The models run on European infrastructure; billing is a monthly flat plan with a token quota. TranslatePilot therefore shows no USD estimate but how many tokens a run is expected to take out of the quota.
- **OpenAI, Google Gemini, Mistral and DeepSeek.** For these providers the base address is built in — you only enter the API key of the respective service. Billing is per token at the provider; TranslatePilot shows the estimated token count per run. A USD limit is possible by entering the price per million tokens in the options — the cost limit then applies too.
- **Custom (OpenAI-compatible).** For any further service speaking the widespread OpenAI interface. Additionally enter the base address of the API, including the version path — for example `https://example.org/v1`.

Choosing a model — the list comes from the provider

Once provider and API key are saved, TranslatePilot queries the available models directly from the provider and offers them in the model field. You always choose from the current offering, without typing model names.

The list appears only **after saving** provider and key — save the options once and reopen them. When the provider is temporarily unreachable, the last saved model value stays selectable; nothing gets lost.

Switching providers

Switching is possible at any time: change the provider, enter the matching key, save, pick the model from the freshly loaded list. Two things are deliberate:

- **The approval applies per provider and model.** After a switch, the first real run requires a freshly reviewed sample — a different model translates differently, and you should have seen that before it writes into your content.
- **Existing translations stay untouched.** The switch alone retranslates nothing; only changed articles get continued with the new provider.

Key security

The API key is used server-side only, never output to the browser and never written to logs.

[Deutsche Fassung](#)

Setting up multilingualism

Before translating, the installation has to work multilingual. The dashboard shows the state; the setup does what is missing — with a free preview, repeatable, and without creating anything twice.

The dashboard: readiness checks

Components → *TranslatePilot* checks on every call:

- Content languages: at least two published?
- Plugin "System - Language Filter" enabled, item associations switched on?
- Site default language: does it match the source language? If not, the target language would sit at the root of the addresses and the source language behind a language prefix — the dashboard warns before the first run creates facts.
- Articles per language and articles set to language "All" (those would be skipped)
- Mapping table and access to the AI provider

The setup step

The setup is **step 3 of the wizard** (*Components* → *TranslatePilot* → *Wizard*); it translates nothing and therefore needs no approval. "Preview (free)" shows what would happen; "Run" does it. In detail:

- Create and publish **content languages** for source and target languages.
- Enable the **language filter and language code** plugins; in the language filter switch on item associations (they link translation and source on save) and drop the source language's URL prefix (existing addresses stay unchanged — important for links and search engines).
- Ensure **one home page per language** — without it the language switcher shows nothing.
- Create or publish **the language switcher**. On YOOtheme sites it is placed in the `navbar` position automatically, because YOOtheme does not render the classic `header` position.
- On request (checkbox): move **articles and categories of language "All"** to the source language. Content set to "All" appears in every language but never gets translated — on a truly multilingual site it belongs to the source language.

Every step first looks at how things stand and skips what already fits. Running the setup repeatedly is explicitly harmless.

Order for the first time

1. Install language packs (*System* → *Install Languages*).
2. In the options set provider, key, model, source and target languages.
3. Open the wizard and follow its steps: setup (preview, then run), translation profile, sample and approval, run — see the next pages.
4. After translating: step 8 of the wizard mirrors menus and modules, so the target language gets its own navigation.

[Deutsche Fassung](#)

The translation profile

The translation profile is the site-specific part of the instructions given to the AI service. It decides whether the translation sounds like your website — or like any website. You find it in the options, tab "Translation profile".

The fields

- **About the website.** Whose website is this? One or two sentences, for example: "Website of an IT service provider for small businesses in the Rhine-Main area."
- **Audience.** Who is this written for? Say: "Owners of small businesses without their own IT department."
- **Tone.** The desired tone of the translation. Empty means: factual and clear, close to the original.
- **Glossary.** Terms that are *never* translated — one per line or comma-separated.
- **Additional instructions.** Free text for everything the fields above do not cover.

Take the glossary seriously

The glossary is for your company name, your product names and third-party brands. Without it the model also translates well-meant things: a company name with a descriptive part becomes an English description, "Backend Booking" becomes a lower-case "backend booking". Everything that is a proper name belongs on the list.

Profile changes retranslate the whole stock

Every change to the profile — even a single glossary entry — makes the **whole stock count as changed**: the next run retranslates all articles, in batches up to the configured limit. That is intended (the old translation was made without the new instruction), but with token-billed providers it costs money. Settle the profile early and change it only deliberately afterwards.

For the same reason the approval is bound to the profile: after a profile change, the next real run requires a freshly reviewed sample.

[Deutsche Fassung](#)

Analysis, sample and real run

The road to the first translation always passes three stations: look at the scope, review and approve a sample, then the run — the **wizard** leads through them in order (steps 5, 6 and 8). After that the run alone suffices, or the scheduled task takes it over. The analysis page shows the same findings at any time as a pure report.

The scope (and the analysis page)

Step 5 of the wizard — and in more detail the analysis page — only reads and shows per target language:

- What is **new**, what has **changed** since its last translation, what is **unchanged** and current, and what gets **skipped** (language "All", excluded articles) — with the reason per article.
- The **cost estimate** for the next batch: in USD for token-billed providers, as tokens against the monthly quota for flat plans.
- **Orphaned translations** — translations whose source no longer exists. They are only reported, never deleted: a source may sit in the trash only temporarily.
- The card "What this run translates — and what not".

The sample

The quality sample translates a few articles (selectable, 1-10) for real through the configured provider but **writes nothing**. Source and translation appear side by side: title, intro, meta description. Review the sample for tone, terminology and glossary compliance — that is exactly what it is for.

The approval

When the sample hits the tone, click "Approve profile". The approval applies to exactly this combination of translation profile, provider and model. When any of them changes, it expires — and the next real run requires a new sample. This way no unreviewed configuration can write into your content, not even through the scheduled task.

The summary: dry run or live run

Step 7 shows every decision on one page — provider and model, languages, profile, scope, approval. From there you start either the **dry run** (shows in full what would happen — free, no API calls, nothing gets written) or the **live run**.

The live run

The real run translates the next batch and writes the translations as regular Joomla articles — with the language association to the source, correct permissions and, where present, transferred field values and tags. Three mechanisms come first:

- **Backup confirmation:** per real run you confirm with a checkbox that a current backup of the site exists.
- **Cost gate:** when the run's estimate exceeds the cost limit from the options, the run aborts before anything is translated. The limit protects against the most expensive accident — an unintended retranslation of the whole stock.
- **Batch limit:** at most the configured number of articles per run (default 20).

Large stocks: just click again

A web run has a time budget. When it runs out, the run stops cleanly — the state is recorded per article, and the next click on "Translate next batch" continues exactly there. Nothing gets lost, nothing gets translated twice.

Continuation

After the initial translation, TranslatePilot detects changes by itself: a changed title, text, meta value or field value makes exactly that article appear as "changed"; the next run continues its translation. Unchanged articles cost nothing.

[Deutsche Fassung](#)

Mirroring menus, modules, footer and templates

Translated articles alone do not make a multilingual website: without its own menu items the target language cannot be navigated, language-bound modules are missing on the translated pages, and YOOtheme footers and templates only know the source language. The mirroring — **step 8 of the wizard** — closes these gaps: each pass with a free preview, each repeatable, each skipping what already fits.

Menu, footer and template mirroring translate texts and therefore stand behind the same profile approval as the articles. The right order: **first translate the articles, then mirror** — twins are only created for targets whose translation exists.

Menu mirroring

Creates menu twins per target language — for every menu item whose target (article or category) is already translated:

- Title and alias of the twin come from the translated content; for category menu items with their own menu title, that title gets translated.
- The twins are sorted into the source language's structure — the navigation of both languages looks the same.
- The home pages are put straight: the real home page becomes home per language, the stand-in home pages created during setup are retired.
- Mirrored source menu items set to language "All" switch to the source language; items without a translatable target stay on "All" and appear in both languages.
- On YOOtheme sites the customizer's menu settings (such as dropdown widths) are transferred to the twins.
- Finally the **link repair** runs: internal links in the translated articles are rewired to the translations, and meta addresses (JSitemap) are moved to the now-real routes.

Module mirroring

Modules set to language "All" appear in every language anyway and stay untouched. Modules pinned to the source language would be missing on the translated pages — the analysis lists them, and the mirroring creates a translated twin per target language: title and content translated, position and visibility taken over, the page assignment switched to the menu twins. Module settings are copied unchanged.

The mirroring only creates; later changes to the source module do not travel into the twin automatically.

Footer mirroring (YOOtheme only)

YOOtheme's footer builder knows no language variants. The way YOOtheme itself documents is builder modules in the `bottom` position, one per language — exactly what this step builds: the footer layout from the customizer is turned into one module per language, the texts translated, internal links rewired. Only once all modules stand is the customizer footer emptied; the old state is saved to a file beforehand.

Template mirroring (YOOtheme only)

YOOtheme renders article detail pages through templates with assignment conditions — and those name the source language's categories. Translated articles therefore fell back to the plain default view. This step creates a copy of each template per target language: categories switched to the translated twins, labels translated, links rewired. Templates whose categories have no translation are skipped.

Without YOOtheme, footer and template mirroring simply report that there is nothing to do.

[Deutsche Fassung](#)

Translating automatically (scheduled task)

After the initial translation, TranslatePilot can take over the daily routine: the bundled task plugin translates new and changed articles on a schedule through Joomla's task scheduler — with exactly the same safeguards as the backend.

Creating the task

1. *System* → *Scheduled Tasks* → *New* and pick the type "**TranslatePilot: Translate articles**".
2. Set an execution rule — say hourly, or nightly.
3. Fill in the task parameters (below) and save.

The task runs the way Joomla intends: through the built-in web trigger, a cron call of `cli/joomla.php scheduler:run`, or the "Test run" button in the task list.

The task parameters

- **User for scheduled runs** (required). Translations are saved under this user when no backend user is logged in. The user needs the permission to create and edit articles.
- **"Regular automatic backups are in place."** The backup confirmation, given per click in the backend, becomes a standing declaration for scheduled runs: as long as this is not consciously set to "Yes", the task does not run.
- **Backup file** (optional) with **maximum age**. When you enter the path of the most recent backup file, the task checks before every run that the file exists, is not empty and is young enough. A silently failed backup job then also stops the translation — instead of writing without a net for weeks.

What the automation does not lower

- **The approval.** Without a reviewed sample of the current profile, provider and model, the task aborts with a clear message. Change the profile and the cron stops until a human approves again.
- **Cost and batch limit** apply per run as in the backend.
- **The mirroring steps** (menu, footer, templates) deliberately never run on a schedule — they are rare structural interventions with a preview, not continuous operation.

Behaviour with large stocks

Every run translates at most one batch and respects a time budget. What no longer fits into the run is taken over by the next one — the state is recorded per article. A large stock is worked off over several runs without anything getting lost.

You find each run's log in the task list at the respective entry; additionally TranslatePilot writes its own log into the Joomla log directory.

[Deutsche Fassung](#)

Frequently asked questions

Why is an article skipped?

The most common reasons stand right next to the article in the analysis: it is set to language "All" (only content of the source language gets translated — the setup can move it), or it is explicitly excluded in the options.

The model list in the options is empty or not loadable

The list comes directly from the provider and needs the saved API key for that: first enter provider and key, save, then reopen the options. If the list stays unloadable, the key is usually invalid or the provider temporarily unreachable — in both cases the saved model value remains selectable.

The real-run button is greyed out

Then the approval is missing: first run and review a sample, then "Approve profile". After every change to profile, provider or model a new sample is due — that is the quality safeguard, not a bug.

What does the translation cost?

That depends on the provider. With token-billed providers (Claude) the analysis shows a USD estimate per batch; typical text articles cost fractions of a cent up to a few cents each. With flat plans (Mittwald AI Hosting) the individual run costs nothing; the tokens count against the monthly quota, and the analysis shows how many. The cost limit per run protects against outliers on top.

A translation was edited by hand — will the next run overwrite it?

Only when the *source* has changed since: then the article counts as changed and gets continued — your manual work on the translation is lost in the process. As long as the source stays unchanged, TranslatePilot does not touch the translation. To maintain a translation by hand permanently, put the article on the exclusion list in the options.

What does the message about orphaned translations mean?

A translation exists whose source was deleted or moved to the trash. TranslatePilot never deletes anything — it reports the finding, and you decide whether the translation stays.

Does retranslating change addresses?

It can: when an existing article is retranslated (say after a profile change), the translation's alias is rebuilt from the fresh title. Internal links are repaired by the link repair; external links pointing at the translation's old address are not. Before planned profile changes on well-linked pages, a look at the most important addresses is worthwhile.

Do I need YOOtheme Pro or JSitemap Pro?

No. Both are optional. With the Joomla core alone everything works: translation, setup, menu and module mirroring. YOOtheme additionally brings builder pages and footer and template mirroring into play; JSitemap Pro its meta overrides.

Does TranslatePilot write while the YOOtheme customizer is open?

It should not: when a run writes into a builder page while the same page is open in the customizer, YOOtheme shows a conflict dialog there. Put scheduled runs outside editorial hours.

[Deutsche Fassung](#)

Changelog

No version of TranslatePilot has been released yet.

[Deutsche Fassung](#)