

Verso

Delivery Specification and Third-Party Components

Document ID	IM.VERSO.INT.05
Document version	1.0
Date of issue	2026-10-06
Applies to	Verso 2026-09-25 (4bac47a)
Language	English
Publisher	iMobile

Contents

- 1. Product identification 3
- 2. Delivery set 3
 - 2.1. Contents 3
 - 2.2. Method of delivery 4
 - 2.3. Documentation 4
- 3. Operating environment 4
 - 3.1. Supported platforms 4
 - 3.2. Hardware and software requirements 4
- 4. Licensing 5
 - 4.1. Editions and terms of use 5
 - 4.2. Activation 5
 - 4.3. Updates 5
- 5. Interface languages 5
- 6. Third-party components 5
- 7. Known limitations 7

About this document

This document defines what is delivered with Verso version 2026-09-25 (4bac47a), the environment the product requires, the licensing terms in functional outline and the third-party software components the product contains.

Prices, delivery dates and the parties' details are not part of this document; they are set in the contract.

1. Product identification

Item	Value
Product	Verso
Version	2026-09-25 (4bac47a)
Release date	2026-09-25
Type of software	Application software: a server-side web application that prepares editable presentations with generative AI models
Deployment	Either installed on the customer's infrastructure or used as a service hosted by the vendor
Publisher	iMobile

Verso builds a presentation from a text prompt or uploaded documents: it drafts an outline, picks slide layouts from a template, writes the text, adds charts, images and icons, lets the user edit the result in the browser and exports it to PPTX or PDF. It is offered as a vendor cloud service paid with credits and as a Docker package installed on the customer's servers under an annual licence.

The application runs in a single Docker container: an nginx front proxy (HTTP, container port 80) routes requests to the web interface (Next.js), the API server (FastAPI) and the MCP server. Data is kept in `/app_data`: an SQLite database by default, or an external PostgreSQL or MySQL database set with `DATABASE_URL`. No language model is included in the image: the application calls an external provider with an API key or a local model server (Ollama, LM Studio, any server with an OpenAI-compatible API). An on-premises installation adds the licence verification module. The cloud service is the same application on the vendor's servers, with sign-in through the vendor site account.

2. Delivery set

2.1. Contents

- On-premises package archive `verso-onprem.tar.gz`: `docker-compose.yml`, `Dockerfile.onprem`, the `verso_license` directory with the compiled licence verification module, the `license` directory for the licence file, `README.md`
- Base application image `verso-production`: built from the application source supplied by the vendor, or supplied pre-built
- Licence file `verso-license.json` carrying the vendor's digital signature
- Operating documentation: product description and functional specification, installation and administration guide, user guide
- Nothing is delivered for the cloud service: access is granted to an account on the vendor site

2.2. Method of delivery

On-premises: links to the package archive and the licence file are shown on the page of the paid order on the vendor site; the base image or the application source is handed over by the vendor as the contract provides. Cloud service: entry from the account area at <https://new-imobile.com>.

2.3. Documentation

The documentation set consists of:

- Product Description and Functional Specification
- Installation and Administration Guide
- User Guide
- Software Lifecycle, Maintenance and Support
- Delivery Specification and Third-Party Components (this document)
- Security and Data Protection Statement
- a country annex for the customer's jurisdiction

3. Operating environment

3.1. Supported platforms

- Server side (on-premises): a host with Docker Engine and Docker Compose v2 that runs linux/amd64 containers; the licence verification module is built for Linux x86_64 and Python 3.11
- User and administrator workstation: a web browser; no client software is installed
- Vendor cloud service: accessed with a web browser from the account area of the vendor site

3.2. Hardware and software requirements

- Docker Engine with BuildKit and Docker Compose v2 (the image build uses BuildKit cache mounts)
- Internet access while the image is built from source (Debian, PyPI and npm packages, components from GitHub); not needed for the build when a pre-built image is supplied
- A free host port for the web interface (5000 in the package) and a TLS reverse proxy if HTTPS access is required
- A text model source: an API key of an external provider or a local model server (Ollama, LM Studio or another server with an OpenAI-compatible API) reachable from the container
- For local models the hardware is determined by the chosen model and model server; the built-in Ollama model list shows file sizes from 0.5 GB to 404 GB
- The vendor does not state minimum CPU, memory or disk figures for the application itself; disk space is used by the image, the data directory and exported files

4. Licensing

4.1. Editions and terms of use

Cloud service: pay-as-you-go credits, spent on generated slides and on messages to the AI assistant.
On-premises: an annual licence (365 days from the date of issue) in the Team (25 users), Business (100 users) and Enterprise (no user limit; the licence file records 100,000) editions. The licence file holds the customer's e-mail address, the edition, the number of users, the list of edition features, the issue and expiry dates and, where used, a binding to the installation. The program checks the signature, the product identifier, the expiry date and the installation binding; the number of users and the edition features are shown in the admin console but are not enforced by the program. After expiry, starting presentation generation is blocked; presentations created earlier remain available.

4.2. Activation

An on-premises installation is activated by placing the licence file in the data directory; there is no online activation. The file is verified locally: the Ed25519 signature against the public key built into the verification module, the product identifier, the expiry date and, if set, the installation binding. Nothing is sent to the vendor. The cloud service needs no activation: the user signs in with the vendor site account.

4.3. Updates

On-premises: the vendor supplies a new image (or source) and package; the administrator rebuilds and restarts the container, and the database schema is upgraded at start-up by Alembic when `MIGRATE_DATABASE_ON_STARTUP=true` is set. The program has no automatic update and no update check. The cloud service is updated by the vendor.

5. Interface languages

User sections are fully translated into English and Russian. Eight more languages (Spanish, Simplified Chinese, Hindi, Arabic, Portuguese, French, German, Japanese) are partly translated, with the remaining labels shown in English. The admin console and the custom template wizard are in English only. The language of a presentation is chosen separately when it is created.

6. Third-party components

Component	Version	Licence	Purpose
Presenton	0.9.3-beta	Apache-2.0	Code base of the product: an open-source project, source snapshot of 2026-08-03; the LICENSE and NOTICE files are included in the image
presenton-export	v0.4.2	не указана / not stated	Pre-built component of the Presenton project for PPTX and PDF export; downloaded when the image is built
FastAPI	0.136.0	MIT	API server framework
Uvicorn	0.45.0	BSD-3-Clause	Python application server
SQLAlchemy, SQLModel, Alembic	2.0.49, 0.0.38, 1.18.4	MIT	Database access and schema migrations

Component	Version	Licence	Purpose
FastAPI Users, pwdlib, argon2-cffi, PyJWT	15.0.5, 0.3.0, 25.1.0, 2.12.1	MIT	Accounts, password hashing, session tokens
cryptography	46.0.7	Apache-2.0 OR BSD-3-Clause	Ed25519 verification of the licence file, token signing
llmai	0.2.8	не указана / not stated	Unified client for calling language model providers
OpenAI Python SDK	2.32.0	Apache-2.0	Calls to OpenAI and to servers with an OpenAI-compatible API, including Ollama and LM Studio
Anthropic Python SDK	0.96.0	MIT	Calls to Anthropic models
Google Gen AI SDK	1.73.1	Apache-2.0	Calls to Google models (Gemini, Vertex AI), including image generation
boto3	1.42.94	Apache-2.0	Calls to Amazon Bedrock
Mem0 (mem0ai)	2.0.0	Apache-2.0	Presentation memory for the AI assistant; stored locally
Qdrant Client	1.17.1	Apache-2.0	Embedded vector store for presentation memory (files in the data directory)
FastEmbed, ONNX Runtime	0.8.0, 1.24.4	Apache-2.0, MIT	Local text embeddings for memory and icon search
BAAI/bge-small-en-v1.5	—	MIT	Embedding model; the weights are baked into the image at build time
spaCy, en_core_web_sm	3.8.14, 3.8.0	MIT	Text processing in the memory subsystem
FastMCP	3.2.4	Apache-2.0	MCP server
pdfplumber, python-pptx	0.11.9, 1.0.2	MIT	Parsing of PDF and PowerPoint files
Pillow	12.2.0	MIT-CMU	Image processing
LiteParse (@llamaindex/litparse)	1.5.2	Apache-2.0	Text extraction from uploaded documents
Tesseract OCR	—	Apache-2.0	Text recognition in scans and images; Debian package
psycopg	3.3.4	LGPL-3.0-only	PostgreSQL driver used for schema migrations
asyncpg, aiomysql, aiosqlite	0.31.0, 0.3.2, 0.22.1	Apache-2.0, MIT, MIT	PostgreSQL, MySQL and SQLite drivers
PostHog Python client	7.13.0	MIT	Installed as a dependency of the Mem0 library; not called by Verso code
Sentry SDK	2.58.0	MIT	Error reporting; inactive unless <code>SENTRY_DSN</code> is set
Next.js	16.2.6	MIT	Web interface framework
React	19.2.6	MIT	User interface library
Konva, react-konva	10.3.0, 19.2.4	MIT	Slide editor canvas
Redux Toolkit, Tailwind CSS, Tiptap	2.8.2, 3.4.17, 2.26.1	MIT	Interface state, styling, text editor

Component	Version	Licence	Purpose
Recharts, Chart.js, Mermaid	2.15.4, 4.5.1, 11.9.0	MIT	Charts and diagrams on slides
mixpanel-browser	2.67.0	Apache-2.0	Usage analytics client; present in the interface bundle, inactive without a project token, which the release does not set
sharp (libvips)	0.34.5	Apache-2.0 (libvips: LGPL-3.0-or-later)	Image processing during export
Chromium	149.0.7827.196	BSD-3-Clause and others	Slide rendering for PPTX and PDF export; Debian package
nginx	—	BSD-2-Clause	Front proxy inside the container; Debian package
Node.js	20	MIT	Runtime for the web interface, document parsing and export
Python	3.11	PSF-2.0	Server-side runtime
ImageMagick	—	ImageMagick	Image conversion; Debian package
Noto, Liberation, Carlito fonts	—	OFL-1.1	System fonts in the image for rendering and export; Debian packages
Inter, Lato, Montserrat, Onest, Oswald, Playfair Display fonts	—	OFL-1.1	Fonts of the built-in templates
Ollama	—	MIT	Optional local model server; not part of the image, connected through <code>OLLAMA_URL</code>

Each third-party component is used under its own licence, named in the table. The table is a human-readable component list; a machine-readable bill of materials (SPDX or CycloneDX) is not part of this delivery.

7. Known limitations

- Presentation content is written by a language model and may contain inaccuracies; the result depends on the chosen model and must be reviewed by a person
- There is no release numbering of its own; the interface does not show a version number
- One container and one server worker process: generation runs as in-process tasks, a restart interrupts running generations (finished slides are kept); there is no horizontal scaling or high-availability configuration
- The licence technically blocks only the start of presentation generation; the number of users and the edition features in the licence file are not enforced by the program
- No multi-factor authentication, no SAML or OIDC sign-in and no LDAP integration; a user cannot change or recover a password without an administrator, who resets it
- Open registration should not be enabled on an on-premises installation; accounts are created by the administrator
- No audit log of user or administrator actions
- No encryption at rest; provider API keys and access keys are stored in clear text; TLS is not part of the container
- Quantitative limits: up to 50 slides per presentation, up to 8 attachments per prompt, files up to 100 MB, slide images up to 5 MB, and by default no more than 3 simultaneous generations per account

- Fonts are not embedded in an exported PPTX: the look of the text on another computer depends on the fonts installed there
- Deleting a presentation does not delete exported files, images or presentation memory entries; there is no export of all data of an account
- Isolated network: building the image needs internet access; fonts from the theme catalogue are loaded by the browser from Google Fonts; the image contains text recognition data for English only; the vendor has not audited the own network activity of third-party libraries (Mem0 in particular)
- Creating a custom template from a PPTX file is designed for models that can analyse images; the result is not guaranteed with other models
- The admin console and the template wizard are in English only; eight of the ten interface languages are partly translated; the Help link leads to the English documentation of the Presenton project
- In the cloud service the model providers are chosen by the vendor; a cloud user cannot connect an own API key for presentation generation
- Credits in the cloud service interface are topped up through a single payment provider that accepts cryptocurrency, or with a promo code

Document history

Document version	Date	Product version	Change
1.0	2026-10-06	2026-09-25 (4bac47a)	First issue