PharmaSUG China 2026 - Paper DS-217

Deep Integration of AI High-Fidelity Document Translation with LLMWiki Knowledge Base

Zuocheng Xiang, Mediwell (Shanghai) Pharmaceutical Technology Co., Ltd.

ABSTRACT

Regulatory submissions, legal contracts, and clinical documentation in the life-sciences industry increasingly require multilingual versions that preserve both semantic accuracy and typographic fidelity. Traditional machine-translation (MT) pipelines discard document structure, resulting in formatting-loss rates exceeding 40%. Computer-assisted translation (CAT) tools rely on 100% manual post-editing, creating an obvious throughput ceiling. This paper presents an AI-Agent-driven translation architecture that unites (1) a four-stage structural pipeline—deep parsing, context-aware translation, structure-level re-rendering, and dual-layer quality assurance; (2) LLMWiki, a translation-native knowledge base that stores structured terminology, transformation rules, style guidelines, and correction cases; and (3) a closed-loop feedback mechanism in which every post-editing correction automatically enriches the knowledge base, creating a self-reinforcing quality flywheel. In production validation across pharmaceutical regulatory packages, the system reduced formatting errors from over 40% to below 8%, raised cross-document terminology consistency from approximately 60% to 95%, and lowered human post-editing revision rates from 40% to 15%. Delivery cycles compressed from weeks to hours for ten-thousand-page projects, while marginal cost trends toward zero as the knowledge base matures.

INTRODUCTION

Global regulatory frameworks now mandate multilingual submission of drug-master files, clinical-study reports, and pharmacovigilance documents. The worldwide translation-services market exceeded USD 56 billion in 2025, growing at a 7.2% compound annual rate, yet the life-sciences segment faces unique constraints: a single new-drug application can generate tens of thousands of pages that must be simultaneously accurate in terminology, identical in format to the source, and fully traceable across versions.

Three structural pain points dominate current practice:

- Formatting loss. Conventional MT outputs plain text; tables lose merged cells, images lose anchor points, headers/footers misalign, and multi-column layouts collapse. Error rates exceed 40% on complex regulatory dossiers.
- Terminology inconsistency. The same drug substance or adverse-event term may appear in three to five different translations across chapters or documents because no systematic terminology constraint is enforced at inference time.
- Knowledge attrition. Terminology spreadsheets are maintained manually, version control is chaotic, and post-editing expertise walks out the door when translators leave. New staff require four to eight weeks to reach baseline productivity.

The consequences are predictable: rework rates above 30% double project costs, delivery delays threaten compliance deadlines, and quality fluctuations erode sponsor confidence in contract research organizations (CROs) and translation vendors.

Existing commercial toolkits—Aspose, Syncfusion, iText—solve parsing but not translation; traditional CAT platforms such as SDL Trados or memoQ enforce terminology yet remain fully manual, with pricing at CNY 0.10–0.30 per character. A ten-thousand-page regulatory package therefore costs six figures and spans months. The core proposition of this work is to deliver SDK-grade format fidelity, LLM-grade semantic understanding, and knowledge-graph-grade continuous evolution at roughly one-tenth of

conventional cost by combining open-source parsing layers, a self-developed structural-mapping engine, and an LLM translation agent orchestrated through a translation-native knowledge layer named LLMWiki.

CORE TECHNOLOGY

The system architecture follows a four-stage pipeline: (1) deep structural parsing, (2) context-aware neural translation with real-time knowledge injection, (3) structure-level re-rendering, and (4) dual-layer quality assurance. Each stage is decoupled through a normalized JSON intermediate representation, allowing independent evolution of parsing, translation, and rendering modules. Figure 1 illustrates the complete pipeline and the feedback loop that feeds post-editing corrections back into LLMWiki.

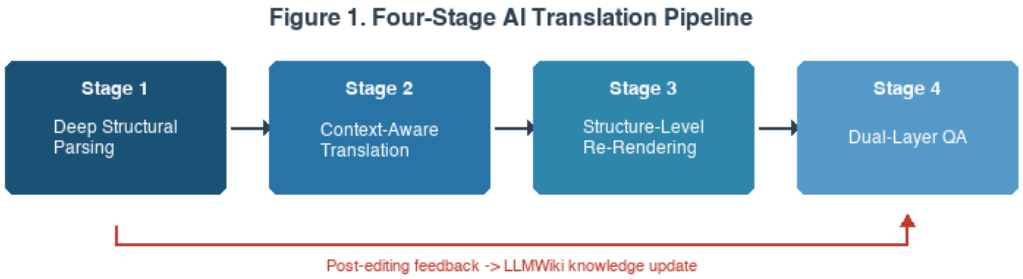

Figure 1. Four-Stage AI Translation Pipeline with Post-Editing Feedback Loop

The pipeline begins with deep structural parsing of the source document into a hierarchical JSON data model. The translation stage consumes text nodes from this model, enriched by real-time retrieval from LLMWiki. Re-rendering reconstructs the target document element by element, preserving original layout and typography. Finally, dual-layer quality assurance combines automated checks with confidence-based human review.

DEEP STRUCTURAL PARSING: FROM FLAT FILE TO HIERARCHICAL DATA MODEL

Regulatory documents are not bags of words; they are nested visual structures. The parser therefore builds a hierarchical tree: Document → Page → Section → Paragraph → Run → Character. Every node receives a globally unique identifier, a bounding-box (bbox) coordinate snapshot, and a complete style record including font family, size, color, bold/italic/underline flags, alignment, indentation, line spacing, and space-before/after values.

Figure 2 shows the hierarchical data model and the properties stored at each node level. Special attention is paid to structures that general-purpose MT routinely destroys:

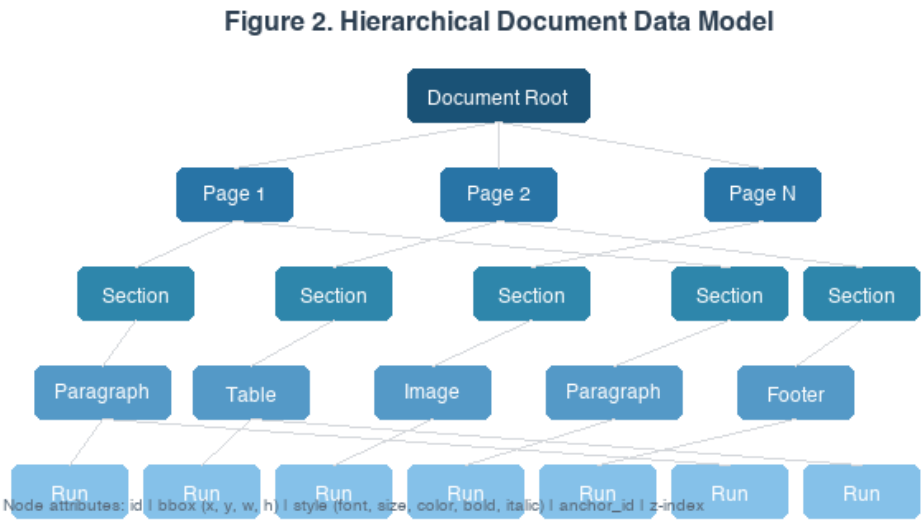

Figure 2. Hierarchical Document Data Model with Node Properties

- Tables. Row and column counts, merged-cell regions, border styles, background colors, and cell-level alignment are all serialized.
- Image anchors. Each image stores its anchor paragraph ID, bounding box, and Z-index so that re-insertion preserves exact positional relationships.
- Footnote and hyperlink reference chains. Bidirectional pointers guarantee that translated footnote text remains connected to the correct callout.
- Style snapshots. Rather than relying on named styles that may not exist in the target environment, the parser inlines all typographic properties, making reconstruction deterministic.

The output is a normalized JSON structure tree that serves as the sole contract between parsing and downstream stages. This decoupling means a new source format—PDF, DOCX, or PPTX—requires only a new parser adapter; the translator and renderer remain unchanged.

STRUCTURE-LEVEL RE-RENDERING: PIXEL-ACCURATE FORMAT FIDELITY

Given the JSON structure tree, the re-rendering engine reconstructs the target document element by element, positioning each node according to its original bbox coordinates and restoring every inlined style property.

Key capabilities include:

- Typography inheritance. Font, color, line spacing, margins, indentation, and alignment are written back verbatim. CJK font auto-substitution is applied when the target locale requires Chinese, Japanese, or Korean glyphs not present in the original font.
- Table exact restoration. Row/column geometry, merged regions, border styles, and background colors are reproduced without loss. Nested tables are supported through recursive tree traversal.
- Human-in-the-loop micro-editing. The rendered output is exposed in a visual editor where post-editors can adjust layout, spacing, or font substitutions. Every manual tweak is logged as a correction case and fed back into LLMWiki.

AI VISUAL UNDERSTANDING: INTELLIGENT HANDLING OF GRAPHICS, CHARTS, AND SEALS

Regulatory dossiers contain dense visual information—flowcharts, organizational charts, chemistry diagrams, and official seals—that cannot be translated as plain text. The system therefore employs a

vision-language model (VLM) zone classifier to categorize each non-text region into one of several classes: vector graphic, raster image, table, flowchart, or seal. Classification accuracy exceeds 95% on internal test sets.

The processing pipeline for visual regions proceeds in three steps:

1. Zone detection. The VLM segments the page into semantic zones and assigns class labels, including support for nested structures such as a table inside a figure.
2. Rasterization. Each detected zone is cropped precisely by its bbox and rasterized at 300 dpi, preserving original coordinates and layer ordering.
3. OCR, translation, and overlay. Text within the rasterized zone is extracted via OCR, translated by the same LLM engine used for body text (ensuring terminology consistency), matched to an appropriate font, and re-rendered onto the original image at the exact same position.

This approach avoids the common failure mode in which MT systems either skip embedded graphics entirely or output disconnected caption text that bears no spatial relationship to the figure.

DUAL-LAYER QUALITY ASSURANCE: MACHINE PRECISION PLUS HUMAN JUDGMENT

Quality assurance is organized into two complementary lines of defense.

The machine defense layer performs four automatic checks:

- Terminology consistency. Each translated term is real-time matched against the LLMWiki terminology table; deviations are highlighted in red with suggested corrections, including synonym disambiguation.
- Omission and duplication detection. Paragraph-level alignment based on edit-distance metrics flags missing, extra, or mistranslated segments.
- Numerical and format validation. Regular-expression rules verify that numbers, dates, serial numbers, currency amounts, and percentages are preserved exactly.
- Format-fidelity scoring. A pixel-level diff between source and rendered pages produces a fidelity score; pages scoring below 92% are automatically flagged for inspection.

The human defense layer focuses cognitive effort where it adds the most value:

- Confidence-based routing. The translation engine attaches a confidence score to every paragraph. Scores above 0.90 pass automatically; scores between 0.70 and 0.90 are sampled; scores below 0.70 are mandatorily reviewed.
- Quality scorecards. Each reviewed document receives a four-dimensional rating—accuracy, fluency, terminology consistency, and format fidelity—enabling trend analysis and translator feedback.
- Feedback reflux. Every post-editing change is automatically extracted as a correction case and written back to LLMWiki, ensuring that the same error never recurs.

The progressive trust model means that as the knowledge base matures, human review effort decreases while quality increases—a rare simultaneous improvement.

LLMWIKI: A TRANSLATION-NATIVE KNOWLEDGE BASE

Generic retrieval-augmented generation (RAG) pipelines retrieve raw text chunks from a vector database and inject them into the LLM prompt. They contain no formatting rules, no terminology constraints, and no style governance. For document translation, this is insufficient: a pharmaceutical submission requires precise drug-name mappings, strict syntax policies, and consistent tone across thousands of pages.

LLMWiki is therefore designed as a translation-native knowledge base with four hierarchical layers, each constraining the translation process more specifically than the last. Figure 3 visualizes the layer architecture, and Table 1 summarizes their content and operational impact.

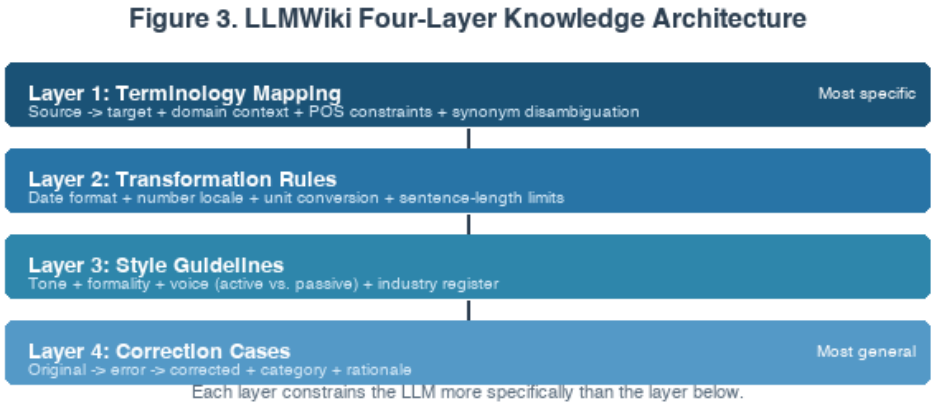

Figure 3. LLMWiki Four-Layer Knowledge Architecture

FOUR-LAYER KNOWLEDGE ARCHITECTURE

Table 1 summarizes the four layers, their content, and their operational impact.

Table 1. LLMWiki Four-Layer Knowledge Architecture

Layer	Content	Impact on Translation
Terminology Mapping	Source-target term pairs with domain context, part-of-speech constraints, and synonym disambiguation rules.	Forces exact term usage; prevents drift across chapters and documents.
Transformation Rules	Syntax-level policies such as date formatting, number localization, unit conversion, and sentence-length limits.	Ensures regional compliance and readability standards.
Style Guidelines	Tone, formality level, voice (active vs. passive), and industry-standard phrasing conventions.	Maintains sponsor brand voice and regulatory register consistency.
Correction Cases	Atomized post-editing feedback: original segment, erroneous translation, corrected translation, error category, and reviewer rationale.	Enables few-shot learning; the LLM sees concrete examples of what not to do.

Knowledge acquisition follows a three-source pipeline: post-editing feedback, automated quality-assurance reports, and external terminology imports (for example, MedDRA, WHO Drug Dictionary, or sponsor-specific master term lists). An LLM-based extraction module converts raw feedback into structured knowledge records, which are then validated by a human curator before promotion to the active knowledge base.

QUANTIFIED IMPACT

Table 2 presents three core metrics before and after LLMWiki deployment in a live regulatory-translation workflow.

Table 2. Quantified Impact of LLMWiki on Translation Quality and Efficiency

Metric	Before LLMWiki	After LLMWiki
Terminology consistency	~60% (uncontrolled drift)	~95% (system-level enforcement)
New-domain cold-start time	Several weeks	Several days
Human post-editing revision rate	~40%	~15%

The 35-percentage-point gain in terminology consistency is achieved through forced alignment against the terminology table combined with RAG real-time injection. The cold-start compression from weeks to days reflects the ability to bulk-import existing term lists and rapidly learn from initial post-editing cycles. The revision-rate reduction of 25 percentage points means that human reviewers shift from repetitive error correction to high-value semantic judgment, effectively raising human efficiency by a factor of 2.5.

CLOSED-LOOP FEEDBACK AND THE QUALITY FLYWHEEL

A translation tool that does not learn from its own output is condemned to repeat the same mistakes on every new project. The system therefore implements an end-to-end closed loop in which every document traverses the full cycle: parse → translate → render → quality-check → post-edit → knowledge update. The post-editing stage is not an endpoint; it is the primary source of new knowledge.

THE POSITIVE-FEEDBACK FLYWHEEL

Figure 4 illustrates the flywheel mechanism. Every translated document generates a volume of text; post-editing of that text produces corrections; corrections are atomized, categorized, and written into LLMWiki; LLMWiki then constrains the next translation task, raising quality and confidence scores; higher confidence reduces the proportion of text requiring human review; reduced review time lowers cost and shortens delivery cycles; faster delivery permits larger document volumes, which in turn generate more corrections and richer knowledge. The result is a self-reinforcing spiral: more translation leads to thicker knowledge, which leads to higher quality, which leads to lower cost.

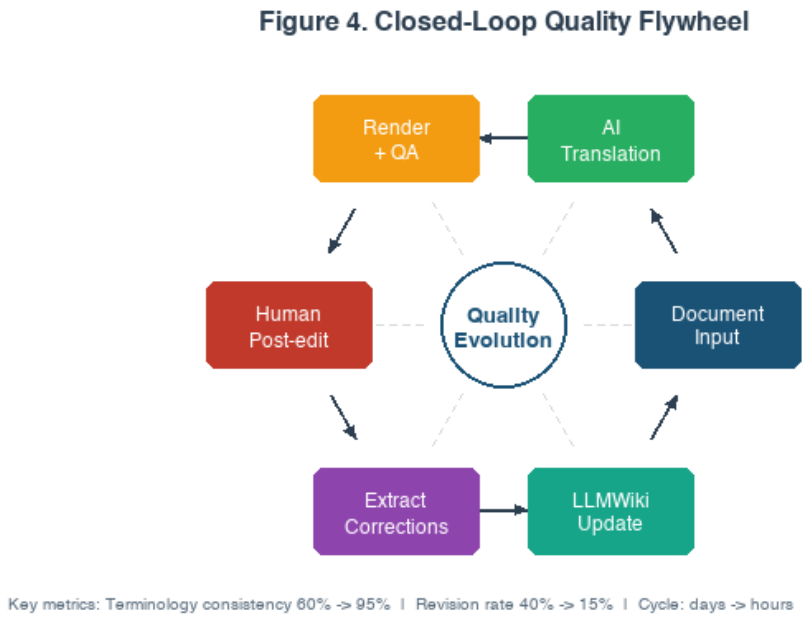

Figure 4. Closed-Loop Quality Flywheel: Document Input to Knowledge Evolution

This dynamic exhibits three evolutionary phases:

4. Cold-start (months 1–2). Post-editing rates exceed 60%; the knowledge base accumulates rapidly from a near-zero baseline.
5. Growth (months 3–6). Auto-pass rates exceed 70%; human effort concentrates on ambiguous or high-stakes passages.
6. Maturity (month 6+). In stable domains, auto-pass rates exceed 90%; the workflow approaches full automation for routine submissions.

Key trend metrics observed during a six-month production pilot in pharmaceutical regulatory translation include:

- Terminology consistency: 60% → 95% (+35 percentage points).
- Post-editing revision rate: 40% → 15% (-25 percentage points).
- Delivery cycle: days → hours (10× acceleration for large packages).

The flywheel therefore converts translation from a repetitive labor cost into a compounding intellectual asset. Each project makes the next project cheaper, faster, and more accurate.

CONCLUSION

AI-Agent-driven structural parsing and re-rendering, when combined with a large-language-model translation engine, simultaneously solves the three-dimensional challenge of semantic coherence, format fidelity, and terminology consistency. Formatting error rates on complex regulatory dossiers fell from over 40% to below 8%.

LLMWiki elevates translation from a disposable labor activity to a continuously evolving knowledge system. Terminology consistency rose from approximately 60% to 95%, and new-domain cold-start time compressed from weeks to days.

The synergy between translation engine and knowledge layer means that quality is not a terminal state but the starting condition for the next translation task. The positive-feedback flywheel drives persistent cost reduction and quality improvement, creating a defensible competitive advantage.

A clear progressive-automation path exists: from heavy human dependence (>60% post-editing) through human-machine collaboration (~30% post-editing) to near-full automation (<10% post-editing) within six months of domain-specific deployment.

In short, high-fidelity translation combined with a knowledge flywheel constitutes a continuous quality-evolution engine for regulated multilingual documentation.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Zuocheng Xiang

Mediwell (Shanghai) Pharmaceutical Technology Co., Ltd.

zuocheng.xiang@mediwell.ai

https://mediwell.ai