Knowledge Organization

Knowl. Organ. **2026**; 53(4): 50562
https://doi.org/10.31083/KO50562

Article

Human-in-the-Loop Generative AI for KOS Registry Publishing: Designing a Dual-Automation Workflow

Ziyoung Park[1,*]

[1]Department of Library and Information Science, Hansung University, 02876 Seoul, Republic of Korea
*Correspondence: zgpark@hansung.ac.kr (Ziyoung Park)
Academic Editor: Natália Tognoli
Submitted: 31 January 2026 Revised: 30 May 2026 Accepted: 4 June 2026 Published: 21 August 2026

Ziyoung Park is a full professor in the Department of Library and Information Science at Hansung University, Seoul, Republic of Korea, where she also serves as Director of the University Library. Her research focuses on knowledge organization systems (KOSs), including the design of classification systems and metadata modeling. She serves as a member of ISKO Italy, an editor for BARTOC, and a program committee member of NKOS. She leads research projects on designing and building registries for Korean KOSs. Her additional work includes developing a bibliographic database of German literature translated into Korean and conducting research on the development of record schedules for managing public records.

Abstract

Knowledge organization system (KOS) registries bring together diverse KOSs under a shared metadata framework, making it easier to gain a coherent overview and compare them across domains and types. However, the contextual documentation needed to properly interpret, differentiate, and interlink individual KOSs is not always readily available or verifiable at the point of registration. This paper treats KOS-related requests for proposals (RFPs) as context-bearing documentation and proposes a human-in-the-loop (HITL), generative artificial intelligence (AI)-assisted workflow for publishing RFP pages in a MediaWiki-based registry. The workflow separates two automations: (1) an Apps Script pipeline that generates MediaWiki-ready wikitext from structured fields in a Notion database through a fixed, rule-based mapping to produce a standardized page skeleton, and (2) slot-based structured summarization that converts narrative RFP text into publishable sections (Tasks, Methods, Deliverables). Accountability is strengthened through expert-led validation of both outputs—verifying the generated page structure and ensuring that slot-based summaries remain faithful to the source RFP text—followed by targeted manual revision where automation alone is insufficient. In an applied case, the structured pipeline was operationally stable, with medium-complexity cases dominated by recurring issues in classification notation formatting and cross-database value mapping. These issues were largely correctable but still required expert review and manual correction. The summarization pipeline produced largely publishable drafts. Revisions were concentrated in the Deliverables slot and in items requiring multiple regeneration runs, while unsupported additions were rare but consequential and were removed or corrected through expert review prior to publication. This study contributes a practical, quality-assured method for publishing KOS-related RFPs as context-bearing registry records by integrating a two-track pipeline—scripted page generation for structured fields and generative summarization for narrative RFP text—with HITL governance for expert verification and correction.

Keywords: knowledge organization systems (KOSs); KOS registries; contextual documentation; requests for proposals (RFPs); human-in-the-loop (HITL) governance; generative AI-assisted wikitext generation; generative AI summarization; MediaWiki

1. Introduction

Knowledge organization systems (KOSs) are widely used knowledge-structuring tools that support the organization and retrieval of knowledge resources across domains. They vary in form and complexity—from term lists and taxonomies to classifications and ontologies—yet share the common function of enabling consistent description, navigation, and interoperability across systems and collections (Gnoli, 2020; Mazzocchi, 2019). These systems are foundational components of knowledge infrastructures, shaping how resources are described and connected for navigation and retrieval.

KOS registries curate metadata in a common schema to provide an overview of and enable comparisons across a heterogeneous set of KOSs by domain and type (Park, 2023). Beyond serving as human-facing inventories for discovery and comparison, registries increasingly function as open-web publishing infrastructures that make KOS resources machine-actionable—discoverable, linkable, and reusable by reference through stable identifiers and hyperlinks—in ways that facilitate linked data publication and semantic interoperability (Voß, 2025; VZG, 2021; Zeng and Mayr, 2019). Accordingly, registries are expected to support core user tasks such as finding, identifying, selecting, obtaining, and exploring KOSs. To support these

tasks effectively, registry records need not only descriptive metadata about the KOS itself but also contextual information, such as its development purpose, design approach, and relevant standards or regulatory constraints (Zeng and Žumer, 2013).

In practice, however, such contextual information is often insufficient when curating individual KOS records. Although some KOSs are accompanied by manuals or introductory materials, many provide little publicly accessible documentation of their development context, and relevant information is often fragmented across external sources. This documentation gap undermines key registry tasks—interpreting, comparing, and selecting KOSs—and constrains linking and reuse across related KOSs. Prior registry-oriented research and practice have made substantial contributions to structuring, categorizing, and exposing KOS metadata for discovery and interoperability (Gnoli et al., 2019; Ledl and Voß, 2016; Park, 2023; Park et al., 2020; Voß, 2016, 2025; Zeng and Mayr, 2019); yet they have addressed contextual documentation less systematically. Building on this foundation, this paper argues that registries should also curate and publish records that document how KOS projects were specified, justified, and evaluated over time. We ground this study in Korean Knowledge Organization System (K-KOS) Open Archives, a MediaWiki-based KOS registry, as an applied setting for publishing such contextual records interlinked with KOS metadata (K-KOS Metadata Project Team, 2025).

As a concrete source of such contextual documentation, we use KOS-related requests for proposals (RFPs), which capture project scope, requirements, constraints, and decision criteria (Nguyen et al., 2022; Park, 2023; Xia et al., 2013). In Korean public procurement, project calls are typically published as a notice webpage accompanied by downloadable attachments, including a detailed RFP document (Uhm et al., 2015). We treat the notice page together with its attached RFP document as an RFP package and curate it as a context-bearing registry record that can be linked to KOS entries. This publication pattern preserves a trace of what the procuring organization requested and how competing proposals were to be assessed, making RFP packages particularly valuable as registry evidence.

This study emerged from a practical need to publish documentary traces of KOS project specifications within K-KOS Open Archives, where RFP packages present a particular challenge: they are heterogeneous and costly to normalize into consistent, registry-ready records. To function as registry entries, RFPs must be published as dedicated MediaWiki pages and explicitly linked to corresponding KOS metadata, which requires both standardized page generation and careful summarization of narrative content. When carried out manually, this work involves extensive manual reformatting and editing and time-intensive drafting, creating a substantial barrier to producing and maintaining RFP pages at scale. For small, research-driven registries, commissioning bespoke automation from specialist developers or vendors is often impractical, underscoring the need for lightweight workflows that domain experts can operate.

Generative artificial intelligence (AI) can shift the balance of effort in registry publishing by lowering the operational burden for domain experts. Rather than directing effort toward engineering-heavy automation or model training, reusable foundation models allow experts to shape outputs by articulating context and criteria through prompts. For narrative sources such as RFP packages, this makes it feasible to normalize heterogeneous text into stable publishable sections while keeping domain judgment central.

Yet registry publishing also amplifies the impact of generative AI limitations, as records are intended to be reused and interlinked on the open web. In this setting, unsupported additions, overgeneration, or subtle misalignments with source documents can introduce quality risks unless outputs are subject to expert review. Accordingly, we use generative AI in a constrained manner, limiting its role to slot-based structured summarization of narrative RFP sections, while keeping structured fields rule-based and scriptable.

For this reason, publishing context-bearing records on the open web requires operational governance that supports expert review, revision, and decision-making over time. Here, we frame human-in-the-loop (HITL) not as data labeling but as governance for publishing: domain experts define constraints, review generated outputs, and approve or revise content where automated processing falls short. On this basis, we design a publication workflow that combines scripted transformation of structured fields with bounded generative summarization of narrative content for MediaWiki publication. Using an applied case from a MediaWiki-based KOS registry, the study evaluates how generative AI and expert judgment can be combined to support sustainable registry publishing and maintenance.

The research questions (RQs) are as follows, with quality outcomes rated as High/Medium/Low:

RQ1. For the structured pipeline (scripted transformation), what quality ratings are achieved, and what recurring mapping and rendering defects account for non-High outcomes?

RQ2. For the narrative pipeline (slot-based structured summarization), what quality ratings are achieved, how is rework distributed across regeneration runs, and which slot-level weaknesses account for non-High outcomes?

RQ3. How does a generative AI–assisted HITL workflow change registry publishing practices relative to manual production and specialist-dependent development?

The remainder of the paper presents the conceptual grounding, the dual-automation workflow and evaluation design, the findings from the applied case, and the resulting implications and limitations.

2. Conceptual Grounding and Design Principles

2.1 KOS RFPs for Discovery, Documentation, and Traces

Public procurement calls often make project documentation publicly accessible through a notice webpage with downloadable attachments. For KOS-related initiatives, these materials can provide a durable documentary trace of what was requested and how proposals were to be assessed, complementing KOS metadata that is otherwise difficult to contextualize (Park, 2023).

Throughout this paper, an RFP package refers to the pair of (1) a project notice webpage (typically containing tabular administrative metadata) and (2) an attached RFP document (typically containing narrative descriptions of project requirements). Conceptually, we treat KOS-related RFP packages as serving three complementary roles in registry work: leads for KOS discovery, surrogate documentation for development and revision context, and documentary traces of legacy, superseded, delayed, or discontinued initiatives. In this study, each RFP package is treated as a context-bearing unit that can be curated and linked alongside corresponding KOS metadata in a registry. For brevity, we use "RFPs" as shorthand for curated RFP packages where no ambiguity arises.

First, RFP packages can function as early, actionable leads for discovering emerging KOS initiatives. Many KOS artifacts become visible to registries only after implementation or public release, whereas procurement notices and attached RFPs provide earlier signals that an organization is initiating a KOS-related project. Registries can therefore create context-bearing records early and later establish KOS links once the resulting KOS can be identified with sufficient confidence.

Second, RFP packages can serve as surrogate documentation when official manuals or other publicly accessible project documentation for a KOS are missing or difficult to obtain. Even when a KOS is implemented, available descriptions often omit the development rationale, constraints, referenced standards, and other contextual factors that shaped the work. RFP packages help fill this gap by preserving stated requirements and procurement-time evaluation criteria for awarding the contract, along with contextual details about scope and expected deliverables. Curating these documents alongside KOS metadata improves interpretability and supports more grounded comparison and linking of related KOSs in registry work.

Finally, RFP packages may remain as durable traces even when a KOS initiative is revised, superseded, delayed, or never fully implemented. In these cases, the package can still be valuable: it documents what was requested and what decision criteria were specified at the time. Preserving such traces in a registry context can reduce duplication of effort, support retrospective analysis of KOS development trajectories, and provide institutional memory of initiatives that shaped the KOS landscape without producing stable artifacts.

2.2 Wiki-Based Registry Publishing on the Open Web

To support discovery, linking, and reuse by reference on the open web, registry publishing imposes a set of functional requirements on how records are made available. Registry records must be discoverable by both human users and external services, citable as stable reference points, and explicitly linked to related resources (Ledl and Voß, 2016; Voß, 2025; Zeng and Mayr, 2019). In addition, because registry descriptions evolve over time, published records must remain correctable and maintainable as links change and new contextual information becomes available.

Wiki-based publishing environments provide a practical way to meet these requirements (Leclercq and Savonnet, 2011; Voß, 2016). Platforms such as MediaWiki support stable page identifiers that function as persistent reference points, rich internal and external linking (e.g., other registries and authority sources), and incremental maintenance through revision histories. Template-based page architectures further enable the consistent rendering of structured fields—such as infoboxes—while allowing narrative, descriptive content to be presented alongside them in the main text. Through this combination, heterogeneous registry records can be published in predictable layouts without sacrificing contextual expressiveness.

Publishing registry records on the open web also extends their value beyond a single registry or community. Because pages are indexable by general-purpose search engines and navigable through internal search and category structures, published records can be discovered, cited, and linked by external users and services. In this setting, reuse primarily occurs by reference: external systems and registries can point to published pages as stable link targets and incorporate them through citation and cross-registry linking, rather than by copying or rehosting content.

Taken together, these requirements highlight the need for publication practices that can reliably render both structured metadata and narrative sources into stable, maintainable web pages, while remaining open to revision and expert verification over time.

2.3 HITL as Governance for Generative AI-Assisted Publishing

In this study, we treat HITL as a governance lens for generative AI-assisted publishing—specifying who holds decision rights over public release and how quality is assured across iterative drafting and revision. In machine learning research, HITL is often framed in terms of human involvement in model development and validation (e.g., labeling and evaluation). These roles, however, also raise a broader question: where human control enters the workflow and how it constrains system outputs. Mosqueira-Rey et al. (2023) describe HITL-ML as an umbrella concept spanning paradigms that differ by where control resides; we adopt this control-centered interpretation because it clarifies how authority and quality checks are exercised when outputs are iteratively generated, reviewed, and revised.

In generative AI–assisted workflows, control is exercised less through direct model modification and more through workflow-level steering of outputs. Practical intervention therefore shifts toward prompt design and structured output formats (e.g., slot- or section-level templates), with humans acting primarily as reviewers who assess, revise, or reject successive drafts. In publishing contexts, this review function must be operationalized, because fluent but weakly grounded text—once released—can persist and accumulate maintenance costs over time.

For registry publishing, we frame HITL not as a post-hoc review but as governance for public release. Governance here means that domain experts retain editorial responsibility for what is published, and that release requires explicit expert review and approval. Accountability is supported through traceable publishing practices: published sections remain re-checkable because the workflow preserves how outputs were produced from specific inputs and prompts (or rules), and how they were subsequently revised during expert review (Papagiannidis et al., 2025). This framing is especially important when generative AI is used to produce narrative text for publication, where outputs may diverge subtly from source documents and therefore require systematic safeguards against unsupported additions. In this way, the approach supports responsible AI practices by making AI assistance transparent, assigning release decisions to designated domain experts, and keeping published text open to verification over time.

2.4 Practical Guidelines for HITL Governance

To operationalize HITL governance in generative AI-assisted registry publishing, it is first necessary to clearly delimit the respective roles of generative AI and domain experts within the workflow. In this study, AI assistance is restricted to tasks that work with explicitly provided inputs: generating and refining MediaWiki-ready scripts based on predefined mapping rules, and producing or regenerating draft summaries based solely on supplied source texts. The workflow explicitly excludes the use of generative AI to create MediaWiki scripts without reference to curated data and mapping rules, or to generate narrative content in the absence of an underlying RFP source.

Second, different treatment is required for structured and narrative fields with respect to disclosure and editorial responsibility. For structured fields, generative AI was used to produce transformation scripts rather than content. The generated Apps Script handled data extraction via application programming interfaces (APIs), field-level mappings from curated databases, and the rendering of MediaWiki page structures. As a result, while mapping errors could occur, the process did not introduce new informational content beyond what was already present in the source data. Narrative fields, by contrast, involved AI-assisted generation of textual content in the form of structured summaries derived from the original RFP documents. Although experts reviewed, regenerated, and selectively modified these summaries, the resulting narrative sections were primarily shaped by AI-generated text, introducing a fundamentally different kind of intervention and justifying explicit disclosure of AI assistance. For this reason, narrative sections warrant explicit disclosure indicating AI assistance and subsequent expert review or modification (e.g., "generative AI-assisted, expert-reviewed"). Such labeling does not function as a technical control, but as an editorial practice that makes responsibility and authorship visible to readers in a registry context where content is intended to be reused by reference.

Third, HITL governance benefits from preserving not only published outputs but also key intermediate artifacts that support review and future maintenance. In addition to the generated wikitext itself, this includes documentation of the original database properties used for generation, the mapping rules that guided scripted transformations, and the scripts employed to produce page content. When scripts are revised during publication or maintenance, retaining versioned records of these changes supports later inspection and correction. Rather than constituting a comprehensive audit log, such retained artifacts provide a practical level of traceability that enables experts to understand how a published page was produced and to intervene effectively when updates or corrections are required.

3. Workflow and Governance Design

This section presents the workflow and governance design for publishing curated RFP packages as linkable MediaWiki pages interlinked with KOS metadata. Although RFP packages are identified and structured upstream (collection from notice portals and curation in Notion), the design and evaluation in this paper focus on the publication workflow—dual-automation page production—and the subsequent expert verification prior to release.

3.1 Data Sources and Curation Workspace (RFP Packages → Notion Evidence Records)

KOS-related RFP packages serve as the primary source material for this study. We identified candidate projects by searching national public notice portals—such as the Public Procurement Service's e-procurement system and the National research and development (R&D) announcement portal—for procurement announcements indicating KOS development or closely related knowledge-organization work. For each candidate project, we collected the public notice webpage together with its downloadable attachments, most notably the RFP document itself. We treat this pair as a single RFP package and use it as the basic evidence unit for registry curation.

To support publication and ongoing maintenance, each RFP package was re-represented in Notion as an internal curation workspace that serves as the master record for both content and operational context. This workspace

adopts a hybrid representation that separates structured and narrative elements while preserving their linkage within a single record. Field-level project metadata derived from the notice are captured as database properties, whereas the narrative body of the RFP is preserved as free text within the corresponding Notion page. Although Notion is not the object of evaluation in this study, it functions as the internal record from which publication-ready pages are derived, refined, and maintained.

Beyond transcription, the curation layer supports expert enrichment. Curators add four types of class numbers, record related internal and external links, and establish explicit relations to KOS metadata entries when the available evidence supports identification. Importantly, RFPs are modeled as standalone evidence records rather than embedded fields within KOS metadata. This design reflects practical registry constraints: some RFPs can be collected without confidently identifying the resulting KOS, while some KOS records lack discoverable RFP documentation. Treating RFPs as independent records allows links to be established conditionally and preserves the distinct roles and lifecycles of implemented KOS artifacts and the procurement documents that specify their intent, requirements, and expected deliverables.

To illustrate the applied workflow, Fig. 1 presents an example of transforming a public procurement notice and its attached RFP document into a MediaWiki-based registry page. The source material combines structured metadata with narrative content distributed across the notice page and the RFP attachment. Through the dual-automation workflow, these materials are reorganized into structured fields and slot-based summaries, making them more consistent, interpretable, and linkable to related KOS entries.

3.2 Publishing Workflows

3.2.1 Baseline: Manual Registry Publishing

In the manual workflow, publishing typically begins by authoring a reference MediaWiki page and duplicating its wikitext to create new pages as templates. After template duplication, structured and narrative components are populated separately.

For structured fields, two manual strategies are common. First, curators copy values from the Notion database property-by-property and paste them into the corresponding template fields, creating repetitive transcription work across entries. Second, curators export Notion data to a spreadsheet and assemble wikitext using spreadsheet functions (e.g., concatenation) by attaching MediaWiki tags to cell values. This approach can accelerate simple field insertion. However, it becomes brittle when complex markup is required (e.g., multi-row tables or nested template structures), where manual correction and troubleshooting are time-consuming.

For narrative RFP content, the source document is available, but curators still need to produce slot-structured summaries from scratch. This involves browsing the document, drafting concise prose aligned with slot boundaries, and converting the text into MediaWiki-ready form. While RFPs share recurring sections, the slot-level rewriting and formatting remain cognitively demanding and difficult to standardize across entries.

Relative to this baseline, the proposed HITL dual-automation workflow replaces manual transcription and from-scratch drafting with two complementary pipelines—structured transformation and narrative summarization—supported by expert review and release decisions (Sections 3.2.2–3.2.3).

3.2.2 Proposed HITL Dual-Automation Workflow

We published KOS-related RFP entries to the K-KOS Open Archives (MediaWiki) using a template-based page architecture. As shown in Fig. 2, the proposed HITL dual-automation workflow separates two complementary pipelines: (i) a structured pipeline that renders field-level metadata into MediaWiki-ready wikitext through scripted transformation, and (ii) a narrative pipeline that produces slot-based summaries from preserved RFP text.

This separation reflects different error profiles and control strategies. The structured pipeline is governed primarily by deterministic mapping and formatting rules, resulting in predictable and identifiable error patterns. By contrast, the narrative pipeline relies on generative summarization and therefore requires bounded regeneration and closer expert verification to mitigate drift and unsupported additions.

Both pipelines output MediaWiki-ready wikitext intended for insertion into a dedicated RFP page template and for explicit linking to the corresponding KOS metadata entry. Expert verification and release decisions integrate the two pipelines at the workflow level, ensuring that both structured and narrative components meet publication standards before public release.

3.2.2.1 Operational Framing: Phases, Rounds, and Runs.

We structured the proposed workflow as two phases, each organized into three operational rounds that reflect how publication work is iteratively prepared, applied, and finalized. These rounds are labeled RS0–RS2 in the structured pipeline and RU0–RU2 in the narrative pipeline (R = round; S = structured; U = unstructured; 0 = initial pilot, 1 = application, 2 = final). Here, a round denotes a distinct stage of coordinated activity within a phase—initial piloting and refinement, controlled application across items, and expert review leading to release decisions. The notion of a run is used only in Phase 2 to describe item-level regeneration attempts during AI-assisted summarization.

In Phase 2, a run corresponds to a single generation of a complete slot-based summary produced under a fixed prompt and slot template. When regeneration is required, multiple runs may be produced for the same item; how-

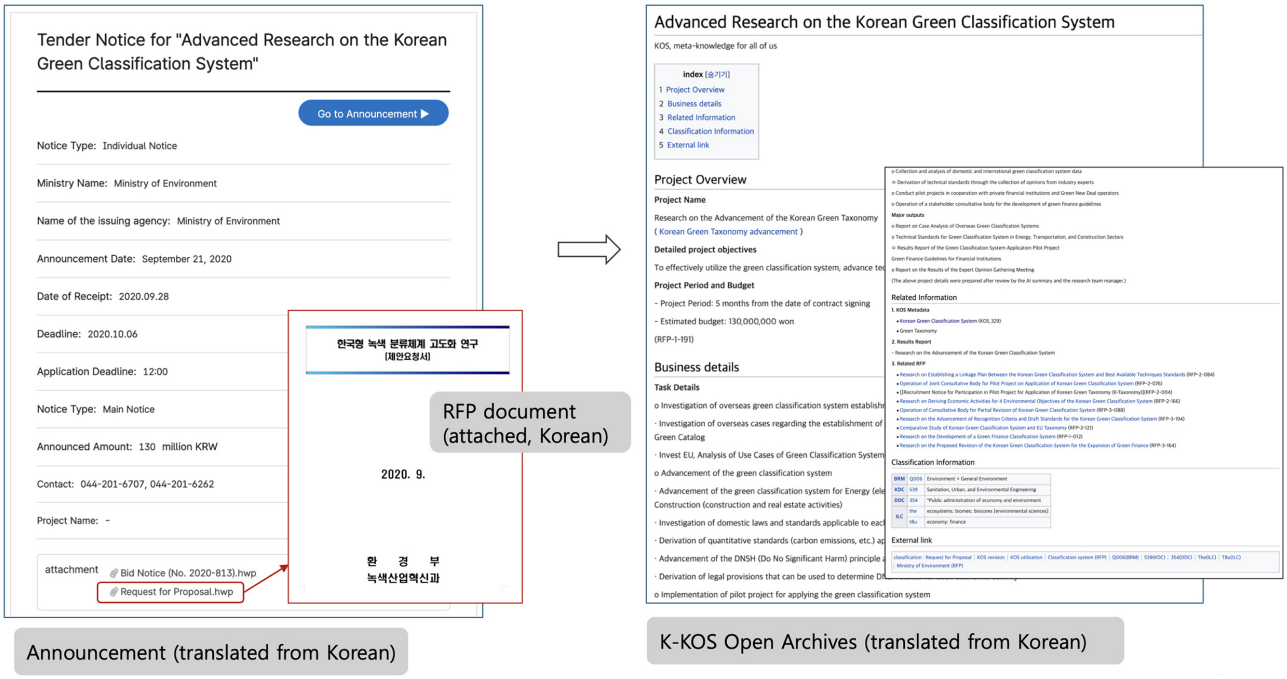

Fig. 1. Example of source RFP notice and corresponding published registry page (K-KOS Open Archives). Source RFP: https://www.ntis.go.kr, Published registry page (K-KOS Open Archives): https://openarchives.net/kos/Korean_Green_Taxonomy_advancement. RFP, request for proposal; K-KOS, Korean Knowledge Organization System.

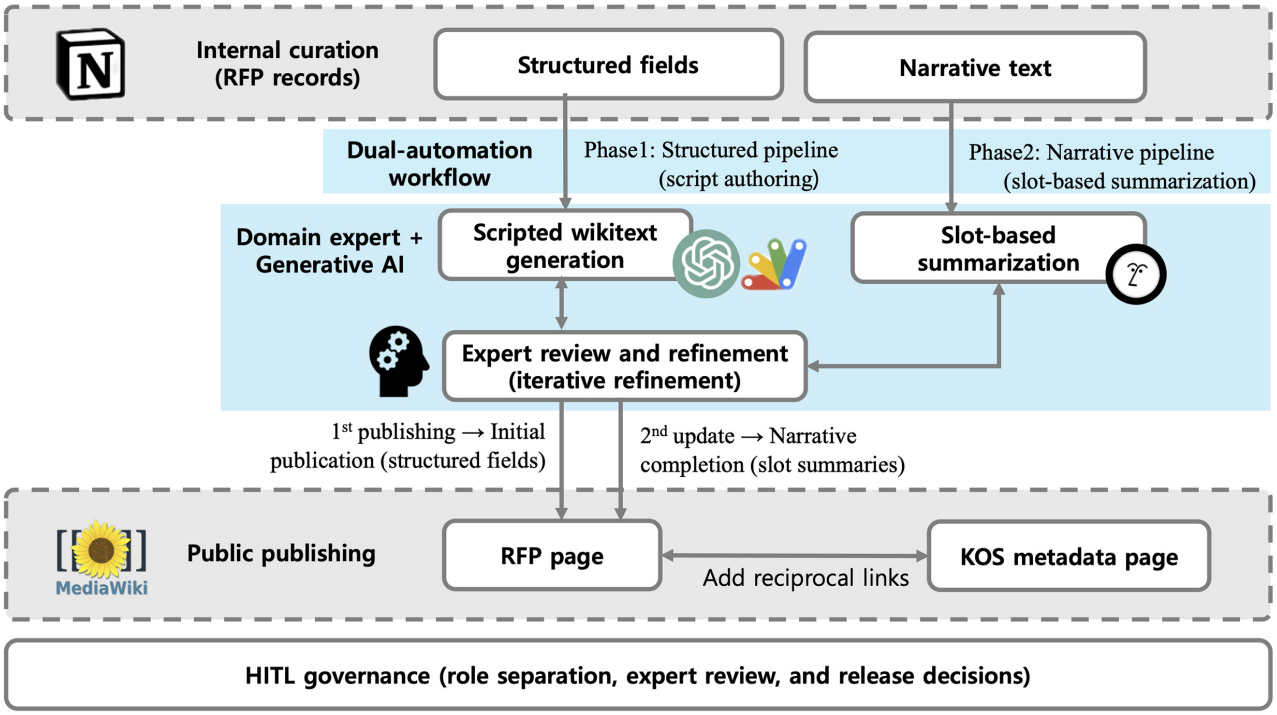

Fig. 2. HITL dual-automation workflow for publishing KOS-related RFP pages in a MediaWiki-based registry. Scripted wikitext generation from structured fields (Phase 1) and slot-based summarization of narrative text (Phase 2), integrated through expert review, refinement, and release decisions. HITL, human-in-the-loop.

ever, evaluation is applied only to the final version retained after expert review and selective revision, not to intermediate drafts. This distinction allows regeneration behavior to be analyzed without confusing intermediate drafts with published content.

For Phase 1 (structured fields), the workflow emphasizes rule stabilization rather than regeneration. An initial pilot round (RS0) is used to refine field mappings and rendering rules based on a subset of items. Once stabilized, these rules are applied deterministically across items in grouped executions (RS1), producing MediaWiki-ready wikitext consistently across all items. A final round (RS2) involves expert review, minor corrections where needed, and publication.

For Phase 2 (narrative fields), an analogous structure is applied with different controls. An initial pilot round (RU0) is used to refine summarization instructions and slot boundaries. A fixed prompt and slot template are then applied across items (RU1), allowing bounded regeneration when summaries fail to meet quality expectations. The final round (RU2) consists of expert review, selective revision, and release decisions.

This phase–round framing makes the workflow repeatable and maintainable: structured outputs can be re-rendered when source data change, while narrative outputs can be selectively regenerated and reviewed without destabilizing already approved content.

3.2.2.2 Phase 1: Structured Pipeline (Scripted Transformation). Phase 1 renders structured properties curated in Notion into MediaWiki-ready wikitext through fixed, rule-based mappings. An Apps Script retrieves item-level values from the internal curation workspace and renders them into a standardized page schema, including an infobox, overview sections, and a Related information section. This section initially lists linked KOS metadata entries, relevant reports, and related RFP records as textual references, which are subsequently verified and, where appropriate, converted into internal wiki links during expert review.

RFP-related metadata were distributed across multiple linked Notion databases, including classification information and managing organizations. While direct field retrieval from the primary RFP record was generally stable, values derived through relational links required additional attention during rule refinement to ensure consistent rendering. Once these mappings were stabilized, structured fields could be rendered deterministically and consistently across all items. Remaining inconsistencies were addressed through targeted expert review prior to publication, as described in Section 3.2.3.

3.2.2.3 Phase 2: Narrative Pipeline (Slot-Based Summarization). Phase 2 produces slot-structured summaries from the preserved narrative text of RFP documents stored in the internal curation workspace. Using a fixed slot template (e.g., Tasks, Methods, Deliverables), AI assistance generates structured draft summaries that are converted into MediaWiki-ready segments for insertion into the target page template. Because narrative quality and slot fit vary across items, this phase allows limited regeneration under a fixed prompt and slot structure, with expert review ensuring that summaries remain aligned with source content and do not introduce unsupported additions.

The slot template was derived from recurring sections observed across the RFP corpus, reflecting how required work and expected outputs are typically specified in procurement documents. Constraining generation to these predefined information elements supports more consistent review and facilitates issue identification and coding in the evaluation stage.

3.2.2.4 Generative AI Configuration. Two different generative AI tools were used in the dual-automation workflow, reflecting the distinct operational roles of the two phases. In the structured pipeline, ChatGPT was used to support Apps Script generation for rule-based transformation of structured fields into MediaWiki-ready wikitext. In the narrative pipeline, Notion AI was used to generate slot-based summaries from preserved RFP text within the Notion curation workspace.

Both tools were accessed through paid subscription services. The experimental work was conducted primarily in July–August 2025. Although exact model version identifiers were not systematically logged in this applied operational setting, the ChatGPT Team plan defaulted to GPT-4o as the primary model during this period. Notion AI does not expose version identifiers to end users; at the time of use, it operated on an internal model configuration without user-facing version disclosure. The summarization prompt template used in Phase 2 is documented in the Appendix, where an initial prompt and four iterative revisions are recorded with their rationale, providing a basis for replication that does not depend on model version.

Prompting was constrained by task type. In the structured pipeline, prompts were used to generate and refine transformation scripts based on predefined field mappings and page structure requirements. In the narrative pipeline, summarization was guided by a fixed slot template and stable instructions designed to extract information only from the supplied RFP text and organize it into predefined sections (e.g., Tasks, Methods, Deliverables).

For Phase 2, regeneration was bounded to a maximum of six runs per item—a practical limit beyond which additional runs yielded diminishing returns and increased the risk of redundancy, semantic drift, and unnecessary review burden.

3.2.3 Expert Verification and Release Decisions

After automated generation, domain experts performed final verification and completion prior to public release. Elements that could not be reliably extracted or generated—such as ambiguous mappings, edge-case formatting, or underspecified narrative content—were resolved through expert review. Release decisions also included confirming required links between the RFP page and

the corresponding KOS metadata record, as well as checking references to external registries where applicable.

This step addresses long-tail cases that are impractical to handle through scripting alone, such as complex table variations or link verification requiring correctness guarantees. Rather than extending automation with exception-heavy rules, these cases are handled through targeted expert intervention, consistent with the governance framing introduced in Section 2.

3.3 Evaluation and Coding

We evaluated outputs using two complementary reporting layers: quality bands (High/Medium/Low; H/M/L) to summarize publishability, and a phase-aligned issue-code taxonomy (S-codes, U-codes, P-codes) to explain sources of rework. This dual reporting supports both distributional reporting (Section 4.1) and phase-specific error-pattern analysis (Sections 4.2–4.3).

3.3.1 Evaluation Scope and Artifacts

The unit of evaluation is an item, defined as one curated RFP package published as a single MediaWiki RFP page. Each item yields two publishable outputs: a scripted wikitext rendering of structured fields (Phase 1) and a slot-based narrative summary (Phase 2). Evaluation was applied to the final retained outputs used for publication after expert review. Because item-level regeneration is used only in Phase 2, run counts are reported exclusively for that phase as a practical indicator of initial summary fitness and the resulting concentration of expert review effort.

3.3.2 Quality Grading Protocol (H/M/L)

We defined three quality bands: H (ready to publish as-is), M (publishable with minor, targeted edits), and L (requiring substantial revision). Band assignment was applied to the retained output for each item, reflecting the level of expert intervention required prior to release. For Phase 2, grading was applied after bounded regeneration under the fixed prompt/slot template, to the final selected output. This protocol provides a consistent acceptability lens across the structured and unstructured components while allowing phase-specific controls to be analyzed separately in Section 4.

3.3.3 Coding Protocol (S-codes, U-codes, P-codes)

Quality bands summarize overall publishability but do not explain the sources of rework. To characterize recurring issues, we applied a phase-aligned coding scheme. S-codes capture recurring issues in the structured pipeline, U-codes capture recurring issues in the narrative pipeline, and P-codes localize narrative issues to specific slots in the summary template. Multiple codes could be assigned to a single item to reflect co-occurring issues. Code definitions and distributions are reported in Section 4.

3.3.4 Evaluation Rationale

This evaluation design is closer to an applied workflow assessment than to model-centric benchmarking — approaches that measure large language model (LLM) capabilities against standardized tasks and metrics. Prior work on large language model evaluation has emphasized that assessment should not be reduced to a single accuracy metric, but should instead reflect multiple dimensions of performance across different task settings (Liang et al., 2023). Surveys of large language model evaluation have also stressed the importance of distinguishing what is evaluated, where evaluation is situated, and how it is conducted across diverse task types and application contexts (Chang et al., 2024). In related domains involving document-oriented and semi-structured specification tasks, a recent study has identified key challenges including hallucination, reproducibility, and interpretability, and have called for evaluation approaches grounded in task context and operational governance (Cheng et al., 2026). The present study follows these principles by using quality bands and issue-based coding to assess operational usability and publishability within a real registry publishing workflow, where source fidelity and expert oversight serve as the primary safeguards against hallucination and unsupported additions in generated outputs.

4. Results and Operational Findings

4.1 Overall Quality Distribution and Operational Signals

This section reports an aggregate view of output quality across the two phases of the dual-automation workflow: (a) Phase 1 scripted transformation for structured metadata and (b) Phase 2 slot-based summarization for narrative text. Outputs were graded using a three-band rubric (High/Medium/Low) aligned with operational rework. For Phase 2, High indicates acceptance without post-editing after no more than three generation runs; Medium indicates either acceptance without post-editing after more than three runs or acceptance after minor post-edits; Low indicates cases requiring substantial revision regardless of run count. High cases carry no issue or slot code annotations; for Medium and Low cases, U-codes and P-codes were used to record reviewer rationales. Fig. 3 presents the resulting quality distributions for both phases.

In Phase 1 (structured), 406 outputs (68.6%) were graded High and 186 (31.4%) were graded Medium, with no Low cases. This distribution indicates that the structured pipeline was operationally stable: once mapping and rendering rules were established, outputs rarely required more than localized correction prior to publication.

By contrast, in Phase 2 (narrative), 485 outputs (81.9%) were graded High, 99 (16.7%) Medium, and 8 (1.4%) Low. Under the run–edit rubric defined in Section 3.3, Medium reflects either iterative convergence (more than three runs without edits) or minor post-edits within

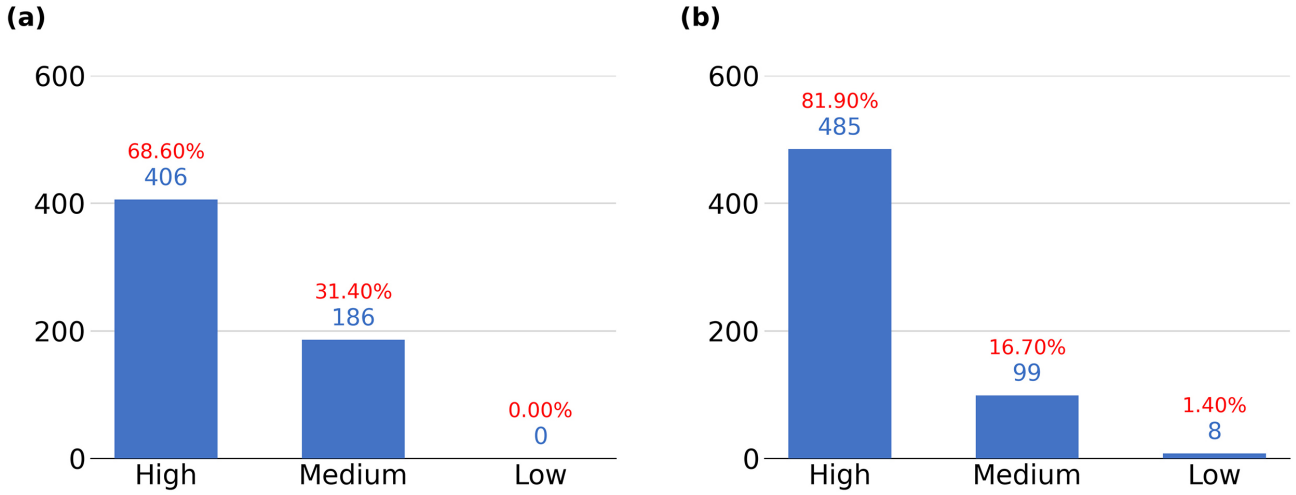

Fig. 3. Quality distribution of outputs by component (N = 592). (a) Phase 1 scripted transformation for structured metadata; (b) Phase 2 slot-based summarization for narrative text. H = High (ready to use), M = Medium (minor edits), L = Low (substantial revision). For Phase 2, Medium and Low cases were annotated with issue codes (U) and slot codes (P); frequency counts are based on non-High cases (n = 107). N denotes the total number of outputs; n denotes a subset.

three runs, whereas Low indicates cases requiring substantial revision regardless of run count.

We analyze phase-specific rework drivers to explain these aggregate outcomes. Phase 1 is executed as a single-pass deterministic transformation after rule stabilization, whereas Phase 2 may require bounded regeneration (Runs ≤6) during slot-based summarization; Section 4.2 reports S-code patterns for structured failures and Section 4.3 reports U-code and P-code patterns (and run statistics) for narrative summaries.

4.2 Phase 1 (Scripted Transformation): Deterministic Issue Patterns

After the mapping and rendering ruleset stabilized, Phase 1 operated as a single-pass scripted transformation, and remaining rework was limited to deterministic markup and normalization fixes. Across 592 structured entries, 406 outputs (68.6%) were graded High and 186 (31.4%) were graded Medium, with no Low cases (Fig. 3a), indicating that Phase 1 rework was dominated by correctable rendering and normalization issues rather than uncertainty in source data or interpretation.

To characterize rework drivers, we applied a phase-specific structured issue-code set (S-codes) to all Medium outputs (n = 186) (Table 1). One issue type dominated: S1 appeared in 185 of the 186 Medium items (99.5%). S2 affected 18 items (9.7%) and always co-occurred with S1, suggesting an edge-case condition tied to formatting or normalization rather than an independent failure mode. S3 was observed in a single item (0.5%). Most Medium cases involved a single issue code (168/186, 90.3%), while the remainder involved two concurrent issues (18/186, 9.7%), indicating highly concentrated and predictable rework patterns.

Qualitatively, recurring issues in Phase 1 were concentrated in a small number of patterned cases rather than dispersed across diverse failure types. The dominant recurring problem involved structural correctness in MediaWiki-ready wikitext, particularly rowspan-related formatting errors. A secondary pattern concerned fixed-width formatting for classification identifiers (e.g., Dewey Decimal Classification (DDC)), where leading-zero padding was occasionally omitted (e.g., “4” instead of “004” or “24” instead of “024”), reflecting normalization requirements during retrieval through relational properties.

Overall, Phase 1 Medium outcomes were dominated by recurrent edge cases tied to wikitext rendering constraints and classification notation formatting or retrieval paths, rather than by unpredictable or content-related failures. These issues recurred in similar forms across items, indicating highly concentrated and patterned sources of rework within the structured pipeline.

4.3 Phase 2 (Narrative Summarization): Bounded Regeneration and Slot-Localized Weaknesses

Phase 2 transforms narrative specifications from the attached RFP document into slot-based summaries (Tasks, Methods, Deliverables). Unlike Phase 1’s single-pass deterministic rendering, Phase 2 permits bounded regeneration (Runs ≤6) to iteratively improve slot-level outputs prior to expert approval.

Across 592 unstructured summaries, High outputs were typically obtained quickly: 373 of 485 High items (76.9%) were produced in a single run. By contrast, runs of three or more accounted for 120 items (20.3%) but 82 of the 107 non-High cases (76.6%), supporting run count as a practical proxy for rework burden in the narrative pipeline (Table 2).

Table 1. Frequency of structured issue codes (S-codes) among Medium items (n = 186).

S-code	Description	Items, n (%)
S1	Rowspan editing error in category name	185 (99.5)
S2	Add leading '0' or '00' to the classification notation	18 (9.7)
S3	Missing DDC classification code	1 (0.5)

Note. S-codes are phase-specific structured issue codes. Percentages are based on Medium items (n = 186). S2 always co-occurred with S1. Issue-code multiplicity: 1 code (168 items, 90.3%); 2 codes (18 items, 9.7%). n, number of Medium items; DDC, Dewey Decimal Classification.

Table 2. Distribution of output quality (H/M/L) by number of generation runs in Phase 2 (N = 592).

Runs	High (%)	Medium (%)	Low (%)	Total (%)
1	373 (99.5)	1 (0.3)	1 (0.3)	375 (63.3)
2	74 (76.3)	23 (23.7)	0 (0.0)	97 (16.4)
3	38 (44.7)	43 (50.6)	4 (4.7)	85 (14.4)
4	0 (0.0)	23 (95.8)	1 (4.2)	24 (4.1)
5	0 (0.0)	8 (80.0)	2 (20.0)	10 (1.7)
6	0 (0.0)	1 (100.0)	0 (0.0)	1 (0.2)

Note. Percentages are computed row-wise, using the number of items finalized at each run count as the denominator. High-quality outputs are concentrated in early runs, while items finalized after four or more runs do not reach High quality. H/M/L, High/Medium/Low.

To characterize why Phase 2 outputs required rework, we assigned phase-specific issue codes (U-codes) to non-High items (n = 107; Table 3a). U2 (slot content misaligned with source or incomplete) was the most frequent issue (93/107, 86.9%), indicating that review effort was dominated by alignment and completeness adjustments rather than missing slot content (U1, 7.5%) or unsupported additions (U4, 6.5%). Source-text corrections (U3) co-occurred in 28 cases (26.2%), reflecting instances where the preserved RFP text itself required normalization or correction during curation.

Finally, we localized weaknesses by assigning slot codes (P-codes) to non-High items (n = 107; Table 3b). P3 (Deliverables) appeared most often (93/107, 86.9%), followed by P2 (Methods) (23/107, 21.5%) and P1 (Tasks) (14/107, 13.1%). This pattern indicates that rework concentrated on deliverables articulation and specification, while tasks and methods were less frequently implicated.

The concentration of rework in the Deliverables slot (P3) likely reflects both source-text heterogeneity and slot-boundary ambiguity. In many RFPs, deliverables are not presented as neatly separable outputs but are embedded in narrative passages that also include functional requirements, scope statements, and performance expectations. This makes extraction less stable than for Tasks or Methods. The result also suggests that the current Deliverables slot may require further refinement, and that slot-based summarization is comparatively less robust when highly specific output requirements are distributed across complex narrative descriptions.

The co-occurrence patterns of U-codes and P-codes across non-High items are summarized in Fig. 4. Medium-grade summaries were dominated by slot-level adjustment needs (U2), most often localized to the Deliverables slot (P3). When source-text correction (U3) co-occurred, remediation involved correcting or clarifying the preserved RFP text before regenerating or editing the summary, underscoring the dependency of narrative summarization on upstream data quality. Unsupported additions (U4) were infrequent but consequential and therefore served as a high-risk signal requiring intensified expert review prior to release.

Low-grade cases more often exhibited U4-type risk signatures— namely arbitrary recombination of fragments and unsupported additions not present in the RFP text. These failures often co-occurred with slot-boundary confusion, where content misassignment produced summaries that appeared well-formed but were poorly supported by the source text. Operationally, regeneration count can serve as a screening indicator: in our dataset, items finalized after four or more attempts never reached High quality, marking a clear operational boundary in regeneration effectiveness. Taken together, the Runs profile and the U-/P-code distributions show that Phase 2 rework is shaped less by formatting errors and more by content alignment and source fidelity, with deliverables extraction as the principal bottleneck.

5. Implications and Limitations

5.1 From Manual Transcription to Review-Centered Governance

At the outset, generative AI was expected to improve efficiency in a conventional sense. As the study progressed, however, it became clear that the workflow was not simply accelerated but reorganized the publishing paradigm by shifting effort from manual transcription and drafting to review-centered controls, bounded regeneration, and expert decision-making at release. The central question is therefore not whether AI saves time, but how it redistributes work across phases. The workflow thus reduces repetitive production work while concentrating expert effort where deterministic correction and narrative judgment remain necessary.

Table 3a. Frequency of unstructured issue codes (U-codes) among non-High Phase 2 outputs (n = 107).

U-code	Definition	Items, n (%)
U1	Missing slot content in the summary (requires addition)	8 (7.5)
U2	Slot content misaligned with source or incomplete (requires adjustment)	93 (86.9)
U3	Source text correction required (RFP)	28 (26.2)
U4	Unsupported content (hallucinated or arbitrary recombination)	7 (6.5)

Note. U-codes characterize recurring failure modes in narrative summarization. Codes are not mutually exclusive; percentages therefore may exceed 100%.

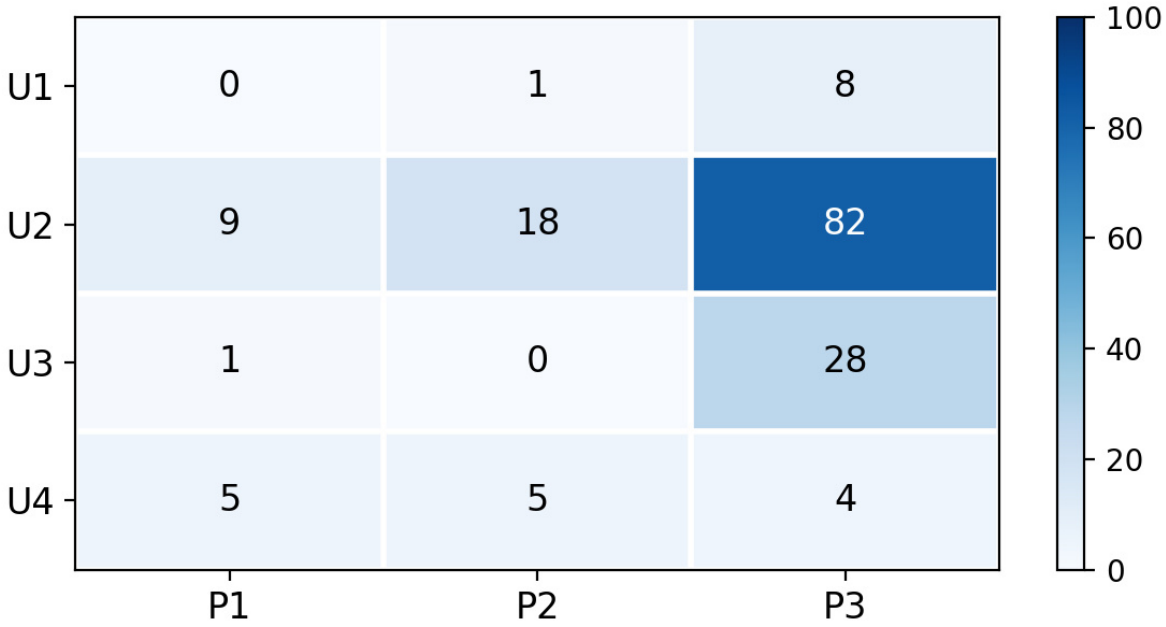

Fig. 4. Co-occurrence of unstructured issue codes (U-codes) and slot codes (P-codes) in non-High Phase 2 outputs (n = 107). Cell values indicate the number of items exhibiting each U–P combination. Multiple U-codes and P-codes may be assigned to a single item; as a result, the same item can be counted in more than one cell, and the sum of cell values exceeds n = 107.

Table 3b. Frequency of slot codes (P-codes) among non-High Phase 2 outputs with slot-localized issues (n = 107).

P-code	Slot name	Items, n (%)
P1	Tasks	14 (13.1)
P2	Methods	23 (21.5)
P3	Deliverables	93 (86.9)

Note. P-codes indicate which summary slots (Tasks, Methods, Deliverables) contained localized weaknesses. Items may receive multiple P-codes.

5.2 Phase-Specific Rework Drivers Require Different Controls

The results show that rework in generative AI–assisted registry publishing is not uniform but strongly phase-specific, with distinct dominant drivers requiring different forms of operational control. In Phase 1, the structured pipeline operates as a rule-governed transformation process. Once mapping and rendering rules are stabilized, remaining issues concentrate in deterministic patterns such as table markup constraints and identifier normalization. These issues are repeatable and predictable across items and are therefore more effectively addressed through validation checks, normalization routines, and schema-level constraints than through item-level intervention. Expert effort in this phase is thus best directed toward stabilizing transformation rules and verifying patterned edge cases at the batch level.

A substantial share of Medium cases in Phase 1 stemmed not from generalizable rule failures but from recurring edge conditions tied to specific values or retrieval paths. Further reduction would have required increasingly specialized scripting beyond what domain experts could sustain; stabilizing rules at a practical threshold and handling residual exceptions through targeted expert review instead proved more effective.

Phase 2 exhibits a contrasting profile. Because Medium includes cases that converged without edits after multiple generation runs, run count should be interpreted as a signal of operational effort and uncertainty rather than as a direct indicator of textual defect. Slot-based summa-

rization is sensitive to source adequacy and to alignment between content and predefined slots, and rework therefore concentrates in iterative cycles of review and regeneration. In this phase, quality depends less on syntactic correctness than on expert judgment regarding fidelity to source content and slot fit.

Repeated runs thus yield diminishing returns and are better treated as signals for earlier expert intervention than as solutions in themselves. When summaries required multiple regeneration attempts, earlier checks of source adequacy or targeted slot-level edits proved more effective than continued regeneration. Practical safeguards in Phase 2 therefore include explicit slot definitions, conservative summarization instructions, and clearly defined expert review and release decisions.

In sum, the dual-automation workflow requires phase-specific forms of control: validation- and normalization-oriented practices in the structured pipeline, and review-centered governance in the narrative pipeline. These differences mirror the observed error distributions, with Phase 1 issues concentrated in a small set of structured S-codes and Phase 2 weaknesses distributed across content-related U-codes, slot-level P-codes, and repeated regeneration.

5.3 Risk Signals and Review Triggers for Sustainable Registry Publishing

This section consolidates the operational implications into two layers: standing guardrails that apply by default and conditional risk signals that trigger intensified review. This separation keeps routine publication lightweight while preserving evidence integrity under human accountability.

Standing guardrails define the minimum rules for routine operation. In the structured pipeline, these guardrails center on deterministic validation, particularly markup integrity and identifier normalization, while long-tail edge cases remain more sustainably handled through targeted expert review than through increasingly complex scripting. In the narrative pipeline, the key guardrail is review-first summarization: slot-based outputs should be treated as structured drafts that require expert confirmation of fidelity to the source and alignment with slot boundaries. Across both phases, accountability should remain explicit through retained regeneration counts, substantial edits, and approval decisions.

Conditional risk signals indicate when routine controls are insufficient. Elevated regeneration counts suggest diminishing returns and should trigger earlier expert intervention rather than further regeneration. Slot-localized weakness, especially in the Deliverables slot, indicates the need for stricter slot-specific checks and more conservative prompting. Inadequate source text likewise calls for conservative editing and heightened review to prevent unsupported additions and preserve a transparent boundary between source evidence and editorial intervention.

Together, these two layers—standing guardrails and conditional risk signals—support a sustainable evidence-publishing model for KOS registries, keeping routine publication reliable while providing practical cues for intensified review when needed.

5.4 Limitations and Future Work

The proposed workflow was developed and evaluated using KOS-related RFPs within the context of Korean public procurement. While its underlying architecture is potentially applicable to other document-oriented domains, its generalizability remains limited in three respects.

First, the applicability of the dual-automation pipeline to non-KOS RFP content has not been empirically tested. Although the workflow may extend to other procurement documents in knowledge infrastructure or research management, its effectiveness is likely to depend on the presence of relatively consistent document structures. More heterogeneous formats may require substantial adaptation of slot definitions and summarization strategies.

Second, the workflow has been validated only on Korean public procurement RFPs. Differences in document conventions, language, and institutional practices across countries may affect both slot-based summarization and structured transformation. While performance may improve in English-language contexts, variations in RFP structure may limit direct transferability.

Third, this study emphasizes operational patterns and quality distributions rather than directly comparable efficiency gains. Because the workflow redistributes effort rather than simply accelerating an unchanged process, conventional "time saved" metrics are difficult to interpret without controlled observational studies. Future work should therefore include cross-domain validation and more systematic measurement of efficiency under comparable conditions.

6. Conclusions

This study presents a HITL dual-automation workflow for publishing KOS-related RFP content in a MediaWiki-based registry. The workflow distinguishes between structured transformation and narrative summarization, assigning each phase different forms of control and review.

The findings confirm that generative AI reorganizes registry publishing rather than simply accelerating it: rework in the structured pipeline is largely deterministic and manageable through validation, while rework in the narrative pipeline requires bounded regeneration and sustained expert judgment.

This study makes three contributions relative to existing KOS registry research. First, it extends the focus from organizing and exposing KOS metadata to curating and publishing contextual documentation that helps explain how KOS projects were specified, justified, and evaluated. In this sense, it builds on prior work on KOS registries and

KOS discovery (e.g., Park, 2023; Zeng and Mayr, 2019) while shifting attention from registry description alone to context-bearing evidence records and their operational publication. Second, it contributes a dual-automation design for registry publishing by distinguishing structured field transformation from narrative summarization, rather than treating registry content as a uniform target of generative AI. Finally, in contrast to prior HITL research that has mainly emphasized human roles in model development, labeling, or evaluation (e.g., Mosqueira-Rey et al., 2023), this study frames HITL as a governance mechanism for publishing, in which human experts function as workflow designers, reviewers, and release authorities.

Overall, the findings suggest that sustainable generative AI-assisted registry publishing depends less on maximizing automation coverage than on clearly delimiting where automation is reliable and where expert judgment must remain decisive. Future work should strengthen automated validation for wiki markup, refine defect taxonomies for narrative summaries, and examine whether the proposed phase-specific controls generalize across document types, procurement contexts, and registry platforms.

Abbreviations

HITL, human-in-the-loop; RFP, requests for proposals; KOS, knowledge organization system; LLM, large language model; DDC, Dewey Decimal Classification; KDC, Korean Decimal Classification; ILC, Integrative Levels Classification; BRM, Business Reference Model; BARTOC, Basic Register of Thesauri, Ontologies & Classifications.

Availability of Data and Materials

The data supporting the findings of this study are available from the author upon reasonable request.

Author Contributions

The author confirms sole responsibility for the conception and design of the study, data collection, analysis and interpretation, drafting and revising the manuscript, and approval of the final version. The author has read and approved the final manuscript and agrees to be accountable for all aspects of the work.

Acknowledgment

Not applicable.

Funding

This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2023S1A5A2A03085177).

Conflicts of Interest

The author declares no conflicts of interest.

Declaration of AI and AI-Assisted Technologies in the Writing Process

ChatGPT (GPT-5.2; OpenAI) was used as an AI-assisted tool for English-language editing during manuscript preparation. Separately, generative AI tools were used within the study workflow itself, as detailed in Section 3.2.2.4 (Generative AI configuration): ChatGPT — which defaulted to GPT-4o on the ChatGPT Team plan during the experimental period — supported Apps Script generation in the structured pipeline, and Notion AI (no publicly available version identifier) generated slot-based summaries in the narrative pipeline. All AI-assisted outputs were reviewed and validated by the author, who takes full responsibility for the study and the manuscript.

Appendix

Summarization Prompt and Iterative Revisions (Phase 2)

The prompts used in Phase 2 were originally written in Korean, as the source RFP documents are in Korean. English translations are provided below for accessibility.

1. Quality guardrails

The following constraints were applied to all summarization runs to ensure fidelity to source documents:

- No unverifiable claims; if a value is unknown, return "unknown" or leave the slot blank.
- Cite only values derivable from the supplied RFP text or mapped Notion properties; never infer budgets or dates.
- Preserve list and bullet structure where present in the source.
- Redact personally identifiable information where required by policy.

2. Summarization prompt and iterative revisions

- **Initial prompt**

> Analyze the text in the current page blocks and produce a concise summary in the exact format below. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script.
> ==사업내용== [Project Description]
> ‘‘‘과업내용’’’ [Scope of Work]
> ‘‘‘수행방법’’’ [Methodology]
> ‘‘‘주요산출물’’’ [Key Deliverables]
> After completing the summary, insert the following line exactly as-is:
> == 이하는스크립트에서제외== [The following content is excluded from the script]

- **Revision 1: Exclude administrative and procedural content**

Added slot-level guidance to distinguish substantive research tasks from administrative and procedural content (e.g., budgets, report copy counts).

Analyze the text in the current page blocks and produce a concise summary in the exact format below. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script. Scope of Work refers to the actual activities carried out in the research project; Methodology refers to the research methods used for those activities — look for content related to surveys, literature reviews, and similar methods. Key Deliverables refers to the principal outputs such as reports and classification schemes.
==사업내용== [Project Description]
‘‘‘과업내용’’’ [Scope of Work]
‘‘‘수행방법’’’ [Methodology]
‘‘‘주요산출물’’’ [Key Deliverables]
After completing the summary, insert the following line exactly as-is:
== 이하는스크립트에서제외== [The following content is excluded from the script]

● Revision 2: MediaWiki code awareness

Added an explicit instruction that the output would be used as MediaWiki script code, and specified that all text in the Notion page block should be rendered in plain text with uniform font size and black font color.

Analyze the text in the current page blocks and produce a concise summary in the exact format below. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script. The output will be used as MediaWiki script code. When presenting in the Notion page block, use plain text throughout, with uniform font size and black font color.
==사업내용== [Project Description]
‘‘‘과업내용’’’ [Scope of Work]
‘‘‘수행방법’’’ [Methodology]
‘‘‘주요산출물’’’ [Key Deliverables]
After completing the summary, insert the following line exactly as-is:
== 이하는스크립트에서제외== [The following content is excluded from the script]

● Revision 3: Attribution phrase added

Added a required attribution phrase to be inserted after the summary.

Analyze the text in the current page blocks and produce a concise summary in the exact format below. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script. The output will be used as MediaWiki script code. When presenting in the Notion page block, use plain text throughout, with uniform font size and black font color.
==사업내용== [Project Description]
‘‘‘과업내용’’’ [Scope of Work]
‘‘‘수행방법’’’ [Methodology]
‘‘‘주요산출물’’’ [Key Deliverables]
After completing the summary, insert the following:
(위사업내용은AI 요약과연구팀담당자의검토를거쳐생성하였음)
[The project description above was generated through AI summarization and review by the research team.]
Leave one blank line, then insert:
== 이하는스크립트에서제외== [The following content is excluded from the script]

● Revision 4: Line-break handling

- Revision 4.1: Line-break handling with custom bullets

The fourth revision addresses the issue where multi-line summaries fail to break lines. When the default bullet style was applied, everything rendered on a single line; “ㅇ” was adopted as a replacement bullet marker.

Analyze the text in the current page blocks and produce a concise summary in the exact format below. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script. Use “ㅇ” as the bullet marker. When presenting in the Notion page block, use plain text throughout, with uniform font size and black font color.
==사업내용== [Project Description]
‘‘‘과업내용’’’ [Scope of Work]
‘‘‘수행방법’’’ [Methodology]
‘‘‘주요산출물’’’ [Key Deliverables]
After completing the summary, insert the following:
(위사업내용은AI 요약과연구팀담당자의검토를거쳐작성하였음)
[The project description above was prepared through AI summarization and review by the research team.]
Leave one blank line, then insert:
== 이하는스크립트에서제외== [The following content is excluded from the script]

**- Revision 4.2: Forced

 for breaks**

Because line-break failures persisted, added explicit

 before each bullet marker to force line separation.

Analyze the text in the current page blocks and produce a concise summary in the exact format below. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script. Use "

ㅇ" as the bullet marker. When presenting in the Notion page block, use plain text throughout, with uniform font size and black font color.
==사업내용== [Project Description]
‘‘‘과업내용’’’ [Scope of Work]
‘‘‘수행방법’’’ [Methodology]
‘‘‘주요산출물’’’ [Key Deliverables]
After completing the summary, insert the following:
(위사업내용은AI 요약과연구팀담당자의검토를거쳐 작성하였음)
[The project description above was prepared through AI summarization and review by the research team.]
Leave one blank line, then insert:
== 이하는스크립트에서제외== [The following content is excluded from the script]

- Revision 4.3: Two-level bullet hierarchy

Introduced a two-level bullet structure ("ㅇ" for primary items, "-" for sub-items) with double
 separators to support more detailed summaries.

Analyze the text in the current page blocks and produce a concise summary in the exact format below. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script. Use "ㅇ" and "

- " as the bullet markers. When presenting in the Notion page block, use plain text throughout, with uniform font size and black font color.
==사업내용== [Project Description]
‘‘‘과업내용’’’ [Scope of Work]
‘‘‘수행방법’’’ [Methodology]
‘‘‘주요산출물’’’ [Key Deliverables]
After completing the summary, insert the following:
(위사업내용은AI 요약과연구팀담당자의검토를거쳐 작성하였음)
[The project description above was prepared through AI summarization and review by the research team.]
Leave one blank line, then insert:
== 이하는스크립트에서제외== [The following content is excluded from the script]

● Revision 5: Exclude budget and project duration

Added explicit instructions to omit budget figures and project duration from the summary, as these are already captured in structured fields.

For each page block in the KOS RFP database where the wiki-exclusion flag (위키제외 (정형)) is "no" and the project announcement ID (사업공고ID) begins with "RFP-", execute the following and insert the result at the top of the page.
Analyze the text in the page blocks below and produce a concise summary in the exact format below. Exclude budget and project duration. Keep the section headings exactly as written (in Korean), because they will be inserted into a MediaWiki script. Use "ㅇ" as the bullet marker, or "ㅇ" and "-" for hierarchical content. When presenting in the Notion page block, use plain text throughout, with uniform font size and black font color.
==사업내용== [Project Description]
‘‘‘과업내용’’’ [Scope of Work]
‘‘‘수행방법’’’ [Methodology]
‘‘‘주요산출물’’’ [Key Deliverables]
After completing the summary, insert the following:
(위사업내용은AI 요약과연구팀담당자의검토를거쳐 작성하였음)
[The project description above was prepared through AI summarization and review by the research team.]
Leave one blank line, then insert:
== 이하는스크립트에서제외== [The following content is excluded from the script]

References

Chang Y, Wang X, Wang J, Wu Y, Yang L, Zhu K, et al. A Survey on Evaluation of Large Language Models. ACM Transactions on Intelligent Systems and Technology. 2024; 15: 1–45. https://doi.org/10.1145/3641289

Cheng H, Husen JH, Lu Y, Racharak T, Yoshioka N, Ubayashi N, et al. Generative AI for Requirements Engineering: A Systematic Literature Review. Software: Practice and Experience. 2026; 56: 141–170. https://doi.org/10.1002/spe.70029

Gnoli C. Introduction to Knowledge Organization. Facet Publishing: London. 2020.

Gnoli C, Park Z, Ledl A. Dimensional Analysis of Subjects: Indexing KOS in BARTOC by Phenomena, Perspectives, Documents and Collections [Conference presentation]. In 1st Low Countries ISKO Conference. Brussels, Belgium. 2019.

K-KOS Metadata Project Team. K-KOS Open Archives. 2025. Available at: https://openarchives.net/kos/K-KOS_Open_Archives (Accessed: 20 January 2026).

Leclercq E, Savonnet M. Structured Wiki with Annotation for Knowledge Management: An Application to Cultural Heritage. International Journal of Digital Information and Wireless Communications. 2011; 1: 264–280.

Ledl A, Voß J. Describing Knowledge Organization Systems in BARTOC and JSKOS. 2016. Available at: https://www.semanticscholar.org/paper/Describing-Knowledge-Organization-Systems-in-BARTOC-Balakrishnan-Ledl/fd6435f8defd32af3e111a4e6d6daecfb29d373c (Accessed: 10 December 2025).

Liang P, Bommasani R, Lee T, Tsipras D, Soylu D, Yasunaga M, et al. Holistic evaluation of language models. arXiv. 2023. https://doi.org/10.48550/arXiv.2211.09110 (preprint)

Mazzocchi F. Knowledge organization system (KOS): an introductory critical account. Knowledge Organization: KO. 2018; 45

Mosqueira-Rey E, Hernández-Pereira E, Alonso-Ríos D, Bobes-Bascarán J, Fernández-Leal Á. Human-in-the-loop machine learning: a state of the art. Artificial Intelligence Review. 2023; 56: 3005–3054. https://doi.org/10.1007/s10462-022-10246-w

Nguyen PH, Lines BC, Maali O, Sullivan KT, Hurtado K, Savicky J. Current state of practice in the procurement of information technology solutions: content analysis of software requests for proposals. International Journal of Procurement Management. 2022; 15: 406–423. https://doi.org/10.1504/IJPM.2021.10038471

Papagiannidis E, Mikalef P, Conboy K. Responsible artificial intelligence governance: A review and research framework. The Journal of Strategic Information Systems. 2025; 34: 101885. https://doi.org/10.1016/j.jsis.2024.101885

Park Z. Aggregating distributed KOSs: enriching with meta information and linking to the multilingual KOS registry. The Electronic Library. 2023; 41: 755–769. https://doi.org/10.1108/el-06-2023-0168

Park Z, Gnoli C, Morelli DP. The Second Edition of the Integrative Levels Classification: Evolution of a KOS. Journal of Data and Information Science. 2020; 5: 39–50. https://doi.org/10.2478/jdis-2020-0004

Uhm M, Lee G, Park Y, Kim S, Jung J, Lee JK. Requirements for computational rule checking of requests for proposals (RFPs) for building designs in South Korea. Advanced Engineering Informatics. 2015; 29: 602–615. https://doi.org/10.1016/j.aei.2015.05.006

Voß J. Classification of Knowledge Organization Systems with Wikidata. In Mayr P, Tudhope D, Golub K, Wartena C (eds.) Proceedings of the 15th European Networked Knowledge Organization Systems Workshop (Vol. 1676, pp. 15–22). CEUR. 2016.

Voß, J. Purpose and properties of the JSKOS data format for knowledge graphs. In SWIB25 conference presentation. Zenodo. 2025.

VZG. Basic Register of Thesauri, Ontologies & Classifications (BARTOC). Verbundzentrale des GBV. Available at: https://bartoc.org/ (Accessed: 6 January 2026).

Xia B, Chan A, Zuo J, Molenaar K. Analysis of Selection Criteria for Design-Builders through the Analysis of Requests for Proposal. Journal of Management in Engineering. 2013; 29: 19–24. https://doi.org/10.1061/(asce)me.1943-5479.0000119

Zeng ML, Mayr P. Knowledge Organization Systems (KOS) in the Semantic Web: a multi-dimensional review. International Journal on Digital Libraries. 2019; 20: 209–230. https://doi.org/10.1007/s00799-018-0241-2

Zeng ML, Žumer M. Managing and Sharing KOS through Registries and RDFa/microdata Using a Metadata Application Profile (Based on the paper of the authors presented in ISKO-UK 2013, with updates of the specification). In ASIST 2013. Montreal, Canada. 2013. Available at: https://nkos.dublincore.org/ASIST2013/Zeng-ZumerASIS2013.pdf (Accessed: 20 January 2026).