ESP-JETA
ESP Journal of Engineering & Technology Advancements
ISSN: 2583-2646 / Volume 5 Issue 4 October 2025 / Page No: 150-155
Paper Id: JETA-V5I4P122 / Doi: 10.56472/25832646/JETA-V5I4P122

Original Article

Bridging the Technical Literacy Gap: Empowering Non-Technical Teams with Curated Code Repositories and Knowledge Transfer Systems

Rupam Priya
Manager - Marketing Analytics, United States.

Received Date: 09 December 2025 Revised Date: 16 December 2025 Accepted Date: 23 December 2025

***Abstract**: The technical literacy gap is leading to significant organizational drag manifested through high volumes of ad-hoc data requests and excessively dependent specialized technical personnel. In response, this paper proposes the Knowledge Transfer System (KTS) - a structured socio-technical approach meant not only to accelerate upskilling but also operationalize organizational learning. The two modern collaboration tools were implemented as support for Externalization and Combination phases of Nonaka's SECI model to realize the goal of turning tacit knowledge into explicit, scalable analytical assets that democratize access to data. It contains a Curated Code Repository (CCR) in GitHub and a Structured Knowledge Base (SKB) on Confluence software. The proposed KTS architecture and Analytical Self-Sufficiency (ASS) empirical frameworks are the major contributions of this work. Validation protocols watch the drop in technical team admin load (ad-hoc requests) and the rise in non-technical user Adoption and Contribution rates. The KTS gives a needed plan for organizational agility, boosting strength in data-heavy settings by moving analytical skill from a focused service to a wide organizational ability.*

***Keywords**: Knowledge Management, Technical Literacy, Collaborative Systems, Analytical Empowerment, SECI Model, Change Management.*

I. INTRODUCTION

Digital transformation initiatives mandate the analytical emancipation of non-technical business units to achieve immediate, self-service, high-quality analytics apart from resource-intensive dependency on specialized data science or engineering teams. These initiatives empower the non-technical operation of business to flow continuously at times of such dependencies. Where this technical literacy gap is not filled, there exists an organizational bottleneck of expert staff unsustainably deluged with ad-hoc data requests and reduced agility within the organization. This has continuously enhanced a dynamic and continuous upskilling solution, which cannot be accommodated within the domain of traditional one-off training, due to rapidly changing technical assets and methodological advancements. The current article responds with the new integrated socio-technical design of the Knowledge Transfer System (KTS) which is engineered both for making tacit technical expertise institutionalized and accelerating enterprise-wide upskilling diffusion. As explicitly architected in KTS, highly validated Curated Code Repository (CCR) on GitHub combined with Structured Knowledge Base (SKB) on Confluence is used to put Nonaka's SECI — Socialization, Externalization, Combination, and Internalization — model of knowledge conversion into operation. It details the core elements of the system, Change Management protocols necessary to sustain adoption, and a rigorous three-dimensional metric framework by which to empirically validate the resultant Analytical Self-Sufficiency (ASS). Its principal contribution is a validated architectural design for converting highly complex developer tacit knowledge into modular, explicit, and scalable self-service analytical assets — a required blueprint for increasing organizational resilience and for getting the foundation of the democratization of data access.

II. THE KNOWLEDGE TRANSFER SYSTEM (KTS) FRAMEWORK AND ARCHITECTURE

KTS is the architectural and procedural implementation of the Nonaka SECI model, essentially those processes that are needed to convert specialized, often tribal technical knowledge into organizational assets available to all. Thus, it raises the solution above simple documentation effort to an involved socio-technical system design problem and addresses the complexity of requirements for architecture and support services that contemporary computing systems require. To understand KTS framework, one needs to understand the two types of knowledge-

- Tactic Knowledge: gained through experience, includes the know-how
- Explicit Knowledge: knowledge based on clear systematic processes that can be well articulated

The SECI Model is what is wanted to happen in an organization for Knowledge Transfer. The acronyms stand for-

- Socialization : from tactic to tactic knowledge
- Externalization : from tactic to explicit knowledge
- Combination : from explicit to explicit knowledge
- Internalization : from explicit to tactic knowledge

A. Theoretical Foundation: Operationalizing SECI for Tacit Knowledge Externalization

Successful KM (Knowledge Management) calls for an organized sharing, retention, and application of knowledge practices that will lead to skills development. The KTS responds to the core imperative of KM which is show how to make explicit knowledge out of the tacit expertise typically held by technical developers — for example, strong familiarity with data structures, optimized scripting, and best practice know how — so that, it can be systematic and available for use by non-experts.

The KTS puts a lot of focus on the Externalization and Combination steps. In Externalization, technical tacit knowledge is carefully expressed as formal artifacts — thorough documentation and standard code pieces. This step asks technical specialists to turn detailed analytical tasks into direct, usable rules, getting rid of the scattered, unspoken hints usually present in basic technical sources. The needed documentation must be very clear and specific since such clarity has been found to work extremely well to share with users who are not experts.

The explicit artifacts are integrated, reorganized, and structured into systematic sets by the Combination role fulfilled by the CCR (Curated Code Repository) and SKB (Structured Knowledge Base). This organizational structure aids in enhanced retrieval of knowledge, maintaining consistency, and enabling cross-functional innovation. The KTS makes sure that knowledge capture is not an 'optional extra' by embedding KM theory as part of the system architecture but is a standardized workflow to be adopted.

B. Architectural Components

The components focused on were-

- Curated Code Repository (CCR)
- Structured Knowledge Base (SKB)

a) The Curated Code Repository (CCR - GitHub)

The CCR does not present itself as a fully integrated environment but rather as an upskilling roadmap consisting of highly validated modular scripts or templates that are parameterized, meant to ensure pre-approved safe-to-use analytical tools for non-technical users. The selection and structuring of the assets in the CCR should strictly follow curation guidelines such that the code snippets demonstrate clarity and high standards in providing prescriptive guidance rather than Accommodation on complex optimization; this emulates rigorous, high-quality code review standards of being applied to ensure utility for non-technical consumption. In order to serve as a learning mechanism, structuring is imposed on CCR by organizing these assets into defined levels of maturity-learning stages (beginner, intermediate, etc.) and technical paths (e.g., certain types of data analysis or report API access). Such structural imposition acts as an architectural constraint supporting knowledge combination by making it systematic and navigable.

Any change or update in the tactic knowledge is updated in the explicit knowledge (CCR), with each change well communicated across all non-technical users of the repository. That is where SKB comes into play.

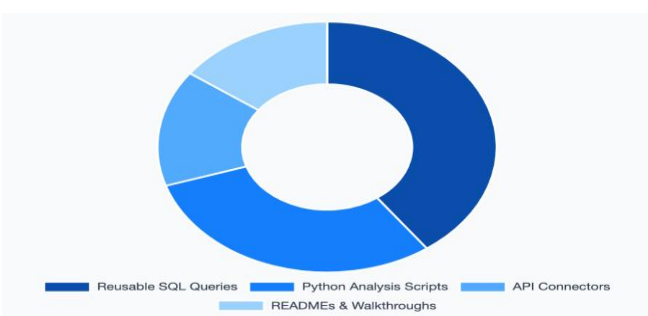

Figure 1 : Curated Code Repository Contents

b) The Structured Knowledge Base (SKB - Confluence)

The SKB provides a context for the KTS, where explicit documentation - such as tutorials, reference guides, and process documents required to bring into context and apply the assets of the CCR can be maintained. The KTS can only function effectively with a disciplined information architecture in SKB. For instance, one major technical decision is the architecture of the Inclusions Library within SKB. Define Inclusion Library. This will standardize reusable content elements — for example common connection strings, key data definitions, or regulatory compliance warnings across several tutorials. This modularity is absolutely necessary for system maintenance and content consistency plus efficient Combination of knowledge since it assures that all derived documents are based on a single, accurate source for common explicit facts.

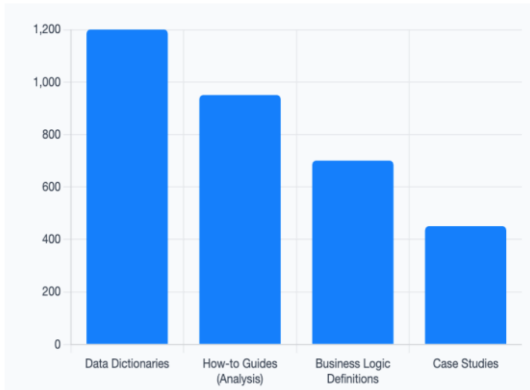

Figure 2: Inclusion Library for SKB

C. KTS Framework Mapping

The KTS framework documents the interaction of the elements of the system with organizational learning theory and shows how the technology enables each of the theoretical stages in knowledge conversion.

Table 1 : The KTS Framework Mapped to the SECI Knowledge Creation Process

SECI Stage	KTS Component	Activity/Mechanism
Externalization	Structured Knowledge Base (SKB)	Systematic documentation of analytical workflows, expert narratives, and 'how-to' guides (Confluence)
Combination	Curated Code Repository (CCR) & SKB	Aggregation of standardized, modular code templates; use of Inclusions Libraries for consistent documentation reuse
Internalization	Training Protocols & Self-Service Use	Non-technical users apply SKB tutorials to execute analytical tasks using CCR scripts, building practical skills

III. IMPLEMENTATION AND CHANGE MANAGEMENT PROTOCOL

The successful shift to Analytical Self-Sufficiency (ASS) depends largely on the strict control of organizational adoption and human factors. For the KTS to work well, its use must go hand in hand with a strong Change Management (CM) plan.

The essential task before implementation of any project is defining the problem to be solved, its scope and assumptions. In this case, defining the end user of the system. Different departments, having dependencies on technical teams to tackle in different areas of business, require KTS setup for different use cases.

A. Change Management and Adoption Strategy

The KTS implementation must directly address the human barriers of resistance and uncertainty, documented as being the main causes of failure in any technology adoption initiative irrespective of its technical prowess. It requires a "people-first" deployment, articulated through continuous communication, targeted training, and visible leadership support toward workforce alignment. Most importantly, organizational theory reminds us that technology adoption is not a static linear process but rather a continuous circular dynamic evolution interrupted by feedback loops and new problems to influence subsequent deployment strategies. This calls for continuous monitoring of user engagement and skill evolution after the launch.

B. Training and Internalization Protocols

The KTS seeks to prompt Internalization, the conversion of explicit knowledge back into tacit knowledge or skill by means of practice. Training protocols should take advantage of empirical evidence on what constitutes effective learning modalities. Therefore, video documentation should be prioritized since users have been seen to significantly and substantially more likely to use visual training material than long textual manuals.Also, the system must work to build a culture of giving back, changing users from passive consumers to active owners. This means setting up rules that make it easy for non-tech users to peer review, check, and suggest changes to the SKB. This spread-out improvement method is key for turning fresh, context-based ideas into clear knowledge, thus keeping assets current, stopping knowledge loss, and maintaining organizational skill. A main design rule here is that having spread thinking ability means there must also be a matching demand for shared writing duty.

IV. EMPIRICAL VALIDATION: MEASURING ANALYTICAL SELF-SUFFICIENCY (ASS)

To validate the KTS architecture and methodology, there has to be quantitative evidence of verifiable workload shift and skill acquisition. The validation framework, therefore, defines Analytical Self-Sufficiency (ASS) in three dimensions: Load Reduction, Adoption, and Contribution.

A. Load Reduction (Primary Economic Metric)

The Weekly Average Number of Ad-Hoc Data Requests received by technical teams is the most definitive measure of KTS efficacy. It directly quantifies the amount of administrative work successfully shifted from high-dollar-value technical talent (e.g., data scientists, engineers) to self-service through the KTS. A large, sustained drop in this metric is core proof of successful KTS deployment and the attained level of ASS. Measuring this requires a rigorously defined baseline data collection period before KTS implementation and then tracking trends long afterward. Translating this administrative time saved into economic return on investment from KTS is just as scheduling or other administrative burdens can be measured.

B. System Adoption

System passive user engagement and utilization of explicit knowledge assets are measured by the frequency of knowledge base logins and viewership of documentation/success rates on searches. High adoption validates that KTS is meeting the organizational requirement for analytical information, hence fulfilling the Change Management strategy in overcoming human resistance and uncertainty. Though internalization is meant to develop skills , sustained adoption becomes imperative because it validates whether the learning path has been effectively used in performing complex or unfamiliar tasks.

C. Active Contribution

Active Contribution will demonstrate the long-term sustainability of KTS and organizational change by knowledge ownership to a shared one. Active Contribution will be measured by the Number of Unique Non-Technical User Contributions, updates, edits, and new documentation submissions to the SKB. This will validate a successful transition process into a collaborative knowledge culture where users continuously ensure not only its quality but also the relevance of knowledge content toward consumption. Therefore, active contribution is a leading indicator for long-term sustainability because it precipitates knowledge retention as well as knowledge currency which are the two major factors that determine the overall success of skills development programs.

D. Metric Framework

Metric Framework Operational metrics as in Table 2 are very important for conducting an empirical study on the success of KTS and later correlating its effect with organizational efficiency and enhancement of analytical potential.

Table 2 : Quantifiable Metrics for KTS Success and Analytical Self-Sufficiency

Metric Category	Quantifiable Indicator	Measurement Rationale
Load Reduction (Efficiency)	Average Weekly Volume of Ad-Hoc Data Requests to Technical Staff	Direct measure of administrative burden mitigation and quantifiable technical staff time savings.
Adoption (Engagement)	Frequency of SKB Logins and Documentation Viewership by Non-Technical Users	Measures system utilization and alignment with organizational objectives; addresses human factors in CM.
Contribution (Sustainability)	Number of Non-Technical User-Generated Content Submissions or Verified Edits	Measures active knowledge flow and the culture of shared ownership, critical for asset currency and

		retention.

While Load Reduction is the ultimate proof of system time savings, Adoption and Contribution metrics are critically required to prove system sustainability. A novelty-induced ad-hoc request reduction may occur temporarily; however, if the SKB is not kept up-to-date and accurate with sustained Contribution, user trust will eventually diminish and finally precipitate a return trip on the Load Reduction metric. Therefore, validated KTS has to require sustained positive trends across all three metric categories that would conclusively prove net benefits of accruing from the developed ASS to the organization.

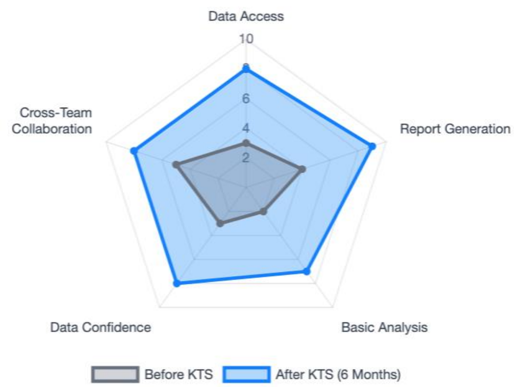

Figure 3 : Impact of KTS in Different Areas

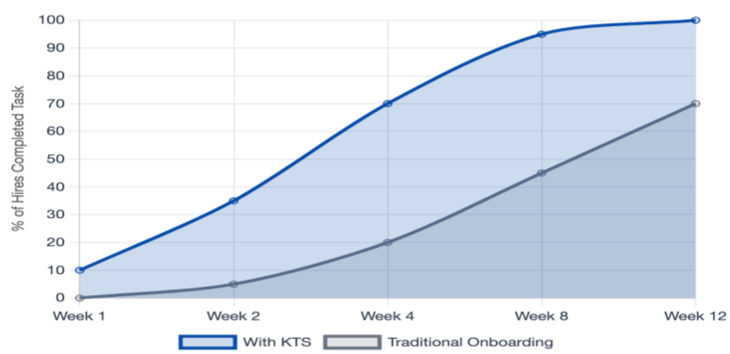

Figure 4 : Impact of KTS in learning curve

This system achieves-

- Analytical Self-Sufficiency: Teams show significant improvement in data confidence, access, and analytical skills post-KTS.
- Faster Onboarding: KTS dramatically reduces the time it takes for new hires to perform their first independent analysis.
- Shift in Ad-Hoc Analysis Source: Over time, the volume of analysis requests submitted to the central tech team decreases as non-technical teams become empowered to find their own answers.

V. CONCLUSION AND FUTURE WORK

The Knowledge Transfer System (KTS) offers a technically rigorous and theoretically grounded approach to bridging the persistent technical literacy gap that continues to challenge organizations. In the process of successfully demonstrating the SECI model through a defined socio-technical architecture- specifically by leveraging both the collaborative strengths and structural constraints of Curated Code Repository (CCR) from GitHub and Structured Knowledge Base (SKB) from Confluence, KTS enables an organized conversion of tacit technical know-how into explicit analytical capability at scale.

The framework requires prescriptive guidance and modularity of content so that explicit knowledge (Externalization and Combination) can be made available to and usable by non-technical teams for fast acquisition of new analytical skills (Internalization). An empirical metric framework proposed herein, based on quantification of Load Reduction, would be a verifiable means whereby the return on investment in knowledge transfer could be determined by the measurable gain in organizational efficiency and validation of Analytical Self-Sufficiency.

Subsequent research should emphasize the application and evaluation of the KTS at different organizational scales and industry domains to test the generalizability of the ASS metric framework in large-scale, longitudinal settings. Another direction for further work is advanced architectural integration involving more sophisticated AI or Large Language Model (LLM) agents. Such agents may support Combination and Internalization dynamically organizing knowledge assets as adaptive instructional interfaces, reducing technical staff involvement in guidance provision and hence even accelerating learning curves for non-technical users.

VI. REFERENCES

[1] Maali, O., Lines, B., Smithwick, J., Hurtado, K., & Sullivan, K. (2020). Change management practices for adopting new technologies in the design and construction industry. Journal of Information Technology in Construction, 25, 325–341. https://doi.org/10.36680/j.itcon.2020.019

[2] Anshari, M., & Hamdan, M. (2022). Understanding knowledge management and upskilling in Fourth Industrial Revolution: transformational shift and SECI model. VINE Journal of Information and Knowledge Management Systems, 52(3). https://doi.org/10.1108/vjikms-09-2021-0203

[3] Billiet, B. (2025, May 22). Top Best Practices for Knowledge Management in 2025 - Whale. Whale. https://usewhale.io/blog/best-practices-for-knowledge-management/

[4] Agile beyond software development: How to empower non-tech teams. (2024, April 18). MiroBlog. https://miro.com/blog/non-tech-agile-strategies/

[5] Adedokun Taofeek. (2024, December 31). Examining the Impact of Knowledge Management on Skills Development Programs. ResearchGate; unknown. https://www.researchgate.net/publication/388063618_Examining_the_Impact_of_Knowledge_Management_on_Skills_Development_Programs

[6] Anthony, J., & Bahati Golyama. (2025). Effects of Employee Self-service on Employee Performance. East African Journal of Business and Economics, 8(3), 72–79. https://doi.org/10.37284/eajbe.8.3.3894

[7] Hugo-Burrows, M. M. (2022). Converting tacit knowledge into explicit knowledge in organisations. Communicare: Journal for Communication Studies in Africa, 21(1), 61–81. https://doi.org/10.36615/jcsa.v21i1.1830

[8] Obrenovic, B., Obrenovic, S., & Hudaykulov, A. (2015). The value of knowledge sharing: impact of tacit and explicit knowledge sharing on team performance of scientists. The International Journal of Management Science and Business Administration, 1(2), 33–52. https://doi.org/10.18775/ijmsba.1849-5664-5419.2014.12.1003

[9] Evans, J. C., Evans, M. B., Slack, M., Peddle, M., & Lingard, L. (2021). Examining non-technical skills for ad hoc resuscitation teams: a scoping review and taxonomy of team-related concepts. Scandinavian Journal of Trauma, Resuscitation and Emergency Medicine, 29(1). https://doi.org/10.1186/s13049-021-00980-5

[10] Zhang, J., Fang, Y., Zheng, H., Fan, S., & Du, T. (2023). The Spatio-Temporal Evolution of Food Production and Self-Sufficiency in China from 1978 to 2020: From the Perspective of Calories. Foods, 12(5), 956–956. https://doi.org/10.3390/foods12050956