

Organizational Models for SAP BW Project Teams in International Offshore and Onshore Delivery Structures

Baris Ozcan*

Freelance Senior SAP BW/BI Consultant and Architect, Dortmund, Germany

Email: bo@bo-consultancy.com

Abstract

This article examines the link between team organization, reporting stability, data reliability, and post-go-live trust in SAP BW environments affected by SAP S/4HANA transformation. Migration programs now unfold under budget pressure, legacy complexity, fragmented ownership, and narrow testing windows. These conditions shift part of project risk from technical conversion into coordination, verification, and decision control across delivery units. The study explains which delivery structures retain control over source extraction, transformation logic, reconciliation, and reporting validation across international offshore and onshore settings. The analytical base combines recent scholarly publications with industry benchmark reports selected for their relevance to ERP post-implementation stability, distributed software collaboration, data governance, and S/4HANA migration pressure. The article develops a conceptual reading of centralized, hybrid, and fragmented team models and formulates practical principles for team design, escalation governance, testing sequence, and monitoring. The proposed interpretation connects delivery structure with the durability of reporting outcomes after go-live.

Keywords: SAP BW; SAP S/4HANA; offshore delivery; onshore delivery; distributed teams; reporting reproducibility; data quality; ERP transformation; testing governance; delivery model.

Received: 4/28/2026

Accepted: 6/28/2026

Published: 7/5/2026

* *Corresponding author.*

1. Introduction

SAP S/4HANA programs are often described as ERP modernization efforts, yet many failures surface outside the visible conversion core. They emerge where long-standing BW data flows, custom extraction logic, reporting dependencies, and inherited master-data weaknesses meet a reorganized delivery structure. In such programs, the migration path explains only part of the outcome. Team configuration shapes the rest because data reproducibility depends on how responsibility moves across BW extractors, transformation logic, reconciliation routines, historical report comparison, post-go-live validation, and business sign-off.

The article aims to determine how organizational models for SAP BW teams influence delivery stability in international offshore and onshore structures during S/4HANA-related change.

Unlike prior studies on ERP transformation, distributed software delivery, and data governance, which typically analyze technical migration risks or coordination practices in isolation [2–7], this study proposes an integrated approach that links organizational team structure directly to reporting reproducibility in SAP BW environments.

The author identifies a research gap at the intersection of SAP BW project organization, offshore/onshore delivery design, and reporting reproducibility during S/4HANA-related transformation. Existing studies discuss ERP migration, BI integration, data governance, or distributed software collaboration as separate research lines, while SAP BW delivery practice shows that reporting instability often emerges precisely where these lines meet. The author addresses this gap by examining how responsibility for extractors, transformations, reconciliation routines, historical report comparison, and business validation is allocated across delivery teams. This analytical focus allows the study to move beyond a general discussion of ERP modernization and to formulate a practical model for BW-heavy transformation programs.

The study pursues three connected objectives. The first identifies the organizational conditions that increase the probability of BW reporting instability during transformation. The second explains how centralized and distributed delivery models differ in their capacity to preserve end-to-end visibility across the reporting chain. The third develops an implementation-oriented model for team design, testing governance, and post-go-live monitoring in projects where BW remains tightly coupled to source-system change.

The novelty of the study lies in combining organizational design analysis with end-to-end reporting validation logic and in treating team structure as a direct explanatory factor for reporting continuity. This integrated perspective has not been systematically articulated in prior research on SAP BW delivery. The working hypothesis states that BW delivery remains more stable when the project model preserves concentrated architectural oversight, explicit responsibility for cross-layer reconciliation, and structured business validation. Fragmented offshore and onshore arrangements extend diagnostic cycles and intensify the effect of data weaknesses and limited testing.

2. Methods and materials

The source base was built as a focused analytical corpus of nine recent publications and benchmark report issued

in 2024. The search targeted major scholarly and professional discovery channels and relied on the following keyword clusters: “SAP S/4HANA migration,” “ERP post-implementation,” “ERP and BI integration,” “data governance,” “distributed agile software development,” “remote software teams,” and “ERP documentation practices.” The selection retained works that addressed one of four questions central to the article: migration pressure in S/4HANA programs, organizational coordination across distributed teams, verification and post-implementation stabilization, and data-governance conditions affecting reporting reliability. Broad discussions of digital transformation without implications for enterprise systems, reporting logic, team coordination, or operational continuity were excluded. The final corpus combines review articles, empirical studies, conceptual work on distributed software delivery, and one benchmark report on S/4HANA migration. The selected literature covers migration constraints, post-implementation success factors, integration governance, data-quality control, meeting and communication practices in remote teams, documentation behavior in ERP projects, and the role of enterprise architecture in distributed coordination [1–9].

The uniqueness of the study design consists in structuring the literature analysis around the full reporting chain, including extractors, transformations, reconciliation, historical report comparison, and post-go-live validation. The author uses this reporting-chain frame to connect scholarly findings with SAP BW/BI project experience, where defects are rarely confined to one technical layer and often become visible only after business users compare current reports with trusted historical outputs. This approach made it possible to capture relationships between organizational design and reporting outcomes that are usually considered separately in existing studies.

The article uses source analysis, comparative interpretation, conceptual synthesis, and analytical typologization. These methods link technical migration risks with organizational design and support the construction of a team-model typology suited to BW-heavy S/4HANA programs. The analytical movement runs from source-based observations to explanatory mechanisms and then to an implementation logic for delivery design.

The key difference from standard literature reviews is the introduction of a cross-layer analytical frame that connects technical data flow elements with organizational responsibility allocation, allowing identification of coordination-induced reporting risks that are not visible in functionally segmented analyses.

3. Results

Recent migration research indicates that S/4HANA programs advance under persistent pressure from cost, compressed timelines, customization complexity, and the need to integrate surrounding systems. SAPinsider reports that data cleansing and quality improvement, custom-code adaptation, and third-party integration remain among the most prominent obstacles for organizations that have already moved, while many firms still face decision pressure as the maintenance deadline approaches [1]. This observation carries special weight for BW-focused projects because analytics and reporting layers are the place where old custom structures, inherited interfaces, and historical logic accumulate. The author underlines that a BW team appears to enter the program late only in formal project plans. In practice, the team is exposed from the first design decision because unresolved ambiguity in source structures returns later as reporting deviation.

The same reports describe migration as a process shaped by technical preparedness and business timing at the same moment. This pattern matters for team design. Organizations that compress scope while preserving daily operations reduce the time available for BW impact analysis, system-fit checks, reconciliation logic, and business validation. Under such conditions, team structure functions as a control mechanism. It either concentrates cross-layer knowledge or distributes it across vendors, functions, and locations [1].

Broader research on digital transformation reinforces this point from another direction. Chang and Chen [2] show that people, process, and technology conditions influence transformation success unevenly, while integration success carries direct performance implications. In SAP BW delivery, technical readiness does not secure reporting continuity when integration work cuts across source applications, transformation rules, and legacy reporting structures. Once integration success is treated as a distinct performance dimension, fragmented team design turns into a measurable project risk instead of a background inconvenience.

A related line of evidence comes from ERP and BI integration research. Chang [3] argues that ERP and BI integration depends on a framework of critical success factors linking organizational conditions to integration success and downstream business performance. In BW programs, this means reporting reliability depends on more than module-level ownership. The reporting chain remains exposed when extraction, transformation, semantic interpretation, and business-facing validation are distributed without a shared control point. A project may stay active across all workstreams and still lose organizational sight of the full reporting path.

The data layer sharpens this argument. Bernardo and his colleagues [4] describe data governance and quality management as a field where value creation depends on disciplined governance practices, assurance logic, and operational control. Acev and his colleagues [5] move closer to the migration problem and note that data migration verification requires the examination of models, schemas, constraints, records, and application architecture, followed by accuracy testing, user training, and post-migration support. Read together, these studies explain why BW teams inherit defects tolerated for years upstream. Weak governance does not remain confined to one table or one process. Project participants carry it through mappings, master data, transformation logic, and historical comparisons until it appears in reports that business users know well enough to challenge.

Organizational form begins to separate project outcomes at this point. Butarbutar and his colleagues [6] identify continuous system integration, post-implementation training, and active user participation among the leading post-implementation factors in ERP success, with organizational and environmental factors outweighing technological ones. In BW delivery, the implication is concrete. A team model that leaves integration responsibility dispersed and keeps business users at the edge of verification delays the moment when semantic defects come into view. By the time the defect is recognized, it has already spread through reconciliations, dashboards, planning inputs, and historical comparisons.

The coordination literature on distributed software work helps explain why such defects remain open for too long. De Andrade and his colleagues [7] report that remote software teams depend heavily on meeting structure, explicit goals, and practices that reduce ineffective collaboration and the erosion of shared understanding. Their review addresses remote software teams in general, yet the implications fit offshore and onshore BW delivery with little

adjustment. Complex reporting logic discussed across time zones and mixed technical-business vocabularies turns each ambiguous meeting into a delayed defect. The later the issue surfaces, the harder it becomes for the team to reconstruct who interpreted a field differently, which mapping changed, or why a technical green status failed to protect business meaning. Alzoubi and Mishra [8] deepen this line of reasoning by showing that enterprise architecture contributes to distributed agile software development as a communication enabler. In BW-heavy S/4HANA programs, architectural oversight performs more than a documentation function. Architects, lead analysts, and responsible delivery managers stabilize interpretation across the chain. A team with concentrated architectural responsibility is better equipped to preserve continuity between source-system change and reporting semantics. A delivery model without such a layer may still close tickets and move transports, yet it lacks a durable mechanism for keeping the whole reporting chain together across organizational boundaries.

Evidence from ERP documentation research points in the same direction. Nakayama and his colleagues [9] show that ERP documentation practices are moving away from archival records toward dialogue-based and transient forms, and that managers who engage with such materials display stronger situational awareness in ERP projects. This matters because BW defects often emerge through interpretation gaps rather than isolated code failures. In a distributed team, situational awareness depends less on the sheer existence of static documents and more on whether the project keeps searchable, decision-relevant records that connect business intent, technical change, and validation logic. If teams fail to preserve that connective layer, offshore handovers narrow, local optimizations multiply, and post-go-live diagnosis slows.

The author uses a comparison across four sources to clarify the organizational mechanism with greater precision. SAPinsider [1] frames migration obstacles through project cost, integration burden, legacy complexity, and data-quality pressure. Butarbutar and his colleagues [6] locate post-implementation stability in integration continuity, training, and participation. De Andrade and his colleagues [7] show that remote collaboration breaks down when teams lack structured communication and explicit meeting practices. Alzoubi and Mishra [8] position enterprise architecture as a communication enabler in distributed delivery. Considered together, these positions support a consistent interpretation relevant to the first two objectives of the article. BW instability during S/4HANA-related change rarely stems from one technical defect. It grows when organizational fragmentation can interrupt end-to-end understanding across the reporting chain, while cost pressure and compressed schedules reduce the opportunities to repair that fragmentation before go-live. The practical distinction between centralized and distributed delivery models follows from the same mechanism. A centralized model preserves interpretive continuity because responsibility for extractor behavior, BW transformations, reconciliation design, historical report comparison, post-go-live validation, and business-facing output remains close to the same control center. A fragmented offshore and onshore model distributes work faster, yet it often separates technical completion from reporting meaning. In such settings, one team confirms transport success, another confirms extractor behavior, and another confirms infrastructure readiness. No one holds the semantic question that determines user trust. The question concerns whether the business can still rely on the number displayed in the report. Research on ERP and BI integration, data governance, and distributed software collaboration converges on this point even though each source approaches it from a different research domain [3–5, 7].

Testing occupies a distinct place in this configuration. The reviewed sources do not describe BW regression in

SAP-specific detail, yet they consistently indicate that post-implementation continuity depends on integration discipline, user participation, and accuracy validation beyond technical completion [2, 5, 6]. In BW projects, the implication is operational. Technical status does not settle the matter because BW defects often remain invisible until end users compare historical report outputs, periods, currencies, master-data groupings, reconciliation totals, or exception logic against familiar reporting behavior. Test governance therefore belongs inside organizational design. Teams cannot leave it to a late-stage technical workstream detached from reporting ownership. Figure 1 presents a conceptual synthesis of the organizational fault line described across the reviewed literature.

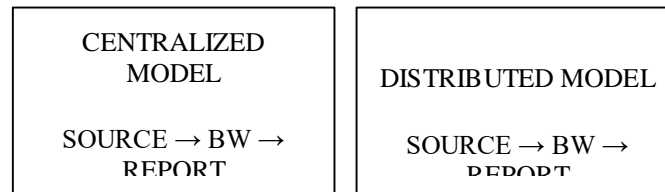


Figure 1: Organizational fault line in SAP BW delivery during S/4HANA transformation (compiled by the author based on [1, 3, 7, 8])

Figure 1 represents a conceptual synthesis rather than an empirical measurement and reflects recurring patterns identified across the analyzed literature. A centralized structure concentrates architectural oversight, reconciliation ownership, and business validation near the BW core. A fragmented offshore and onshore structure disperses those functions across workstreams, which increases the probability that technical completion and reporting meaning diverge.

Across the reviewed sources, the author identifies a consistent analytical pattern: BW-related project resilience depends on whether the delivery model preserves three forms of continuity at the same time, namely continuity of interpretation, continuity of validation, and continuity of ownership. A distinctive contribution of this study is the identification of three interdependent forms of continuity—interpretive, validation, and ownership continuity—as a unified explanatory mechanism for reporting stability. This triadic model has not been explicitly formulated in prior literature on ERP or BI delivery. Where teams preserve these continuities, organizational distance remains manageable. Where teams break them, offshore and onshore structures can increase the risk of defects through routine coordination processes. The analysis indicates that organizational fragmentation is associated with reduced reporting continuity and longer defect-resolution cycles in SAP BW environments during S/4HANA transformation programs. For the first time in the reviewed literature, the analysis identifies a consistent relationship between organizational fragmentation and delayed detection of reporting defects in SAP BW environments, showing that coordination structure itself can act as a primary source of reporting instability rather than a secondary operational factor.

4. Discussion

The reviewed literature supports a stricter interpretation than standard project-method descriptions usually offer.

The interpretation proposed in this study differs from standard project management perspectives by positioning semantic accountability, rather than task execution, as the central organizing principle of BW delivery. The author notes that this shift is derived from practical SAP BW/BI delivery conditions, where technical completion does not guarantee that business users will accept the resulting reports as reliable. This shift allows a more precise explanation of why technically successful projects may still fail at the reporting level. The central design question in BW-heavy S/4HANA programs concerns the location of semantic accountability after work has been split across delivery units. Reporting reproducibility changes the way project structures should be judged. Some models appear cheaper only at the task level. Some models appear slower because they keep ambiguity visible instead of pushing it downstream into reconciliation and user complaints. The obtained results require clarification in relation to previous studies. Prior research links ERP transformation outcomes with people, process, technology, data governance, user participation, and distributed-team coordination [2–8]. The present study specifies this discussion for SAP BW delivery by showing that reporting instability emerges when technical execution is separated from responsibility for business meaning. Unlike studies focused on ERP success factors or distributed collaboration in general, this article connects team structure with source-to-report continuity, reconciliation quality, and post-go-live trust.

A workable implementation model begins with a compact onshore control core. This core keeps ownership of BW architecture, source-to-report lineage, defect triage, reconciliation design, and business sign-off logic. Offshore capacity remains useful for development, routine model adjustments, controlled remediation work, and repeatable transformation tasks. The arrangement becomes less stable when the offshore segment receives technically precise tasks without access to the interpretive logic behind them. The same arrangement breaks down when the onshore side keeps decision power yet fails to preserve the records, exemplars, and review cadence required for precise handover. The first comparison distinguishes delivery forms through the organizational conditions that sustain stability. Table 1 presents a conceptual comparison of SAP BW delivery models based on the reviewed literature.

Table 1: Comparative characteristics of SAP BW delivery models in S/4HANA-related programs (compiled by the author based on [1, 6–8])

Criterion	Centralized model	Hybrid onshore-offshore model	Fragmented distributed model
Architectural ownership	Concentrated and visible	Shared but formally governed	Diffused across vendors and functions
Defect diagnosis speed	Higher in cross-layer issues	Moderate, depends on escalation design	Often delayed by handovers
Reporting semantics control	Strong	Uneven	Weak
Communication burden	Lower	Medium	High
Exposure to requirement drift	Lower	Moderate	High
Testing coordination	Easier to align with business validation	Possible with strong gatekeeping	Frequently split from business meaning
Cost transparency	High	Medium	Often distorted by rework and correction cycles
Post-go-live stabilization	More predictable	Dependent on retained expertise	Frequently prolonged

Table 1 does not imply that centralized delivery is always preferable. Its advantage appears mainly in BW-heavy programs where historical report comparability, master-data consistency, and business validation remain sensitive. A hybrid model can be equally stable when architectural ownership, escalation rules, and validation authority are clearly retained. Fragmented delivery becomes risky when ticket completion replaces shared responsibility for reporting meaning.

This comparison points to one central issue. Delivery models differ less by geography than by their ability to keep semantic accountability intact. A hybrid structure can perform well when escalation rules remain sharp, validation authority stays local and visible, and architectural oversight is not broken into fragments. A fragmented model tends to replace explanation with status reporting. Teams then stop learning from defects and begin routing them across organizational borders.

Testing logic requires separate treatment because many BW failures surface after technical completion has already been declared. A stable program needs a layered test sequence. The first layer checks extractor behavior and transformation correctness. The second verifies reconciliation logic between source structures and BW outputs. The third compares current results with trusted historical reports and expected business totals. The fourth covers post-go-live validation during early stabilization. This sequence consumes time and attention, yet the alternative leaves the program at go-live with unresolved interpretive debt. The second comparison translates this logic into an operational monitoring framework. Table 2 presents a conceptual monitoring framework derived from recurring themes in the reviewed literature.

Table 2: Monitoring metrics for BW team governance in international delivery structures (compiled by the author based on [3–5, 9])

Monitoring dimension	Indicative metric	Why it matters
End-to-end ownership	Share of incidents with one accountable owner across source, BW, and reporting layers	Reveals whether the chain is governable
Reconciliation discipline	Rate of defects detected through source-to-report comparison before UAT completion	Indicates whether testing reaches reporting semantics
Master-data exposure	Number of open BW-impacting issues linked to Business Partner, material, or mapping inconsistencies	Tracks hidden upstream instability
Handover quality	Percentage of offshore tickets reopened after onshore review or business validation	Captures communication loss and rework
Documentation usefulness	Proportion of defects with searchable decision history and mapping rationale	Measures situational awareness, not document volume
Validation latency	Time from technical fix to business acceptance	Shows the gap between system status and trusted output
Post-go-live trust	Number of user-raised report credibility issues during stabilization period	Captures reputational risk in analytics

These metrics matter because they measure organizational behavior at the point where technical and business meanings meet. Traditional project dashboards often count transports, tickets, and test cases. Those indicators still have operational value, yet they explain little when users reject numbers that a project team has already marked as correct. The proposed set redirects attention toward accountability, reconciliation, interpretive continuity, and trust formation.

A practical sequence follows from this discussion. Early in the program, the project team should map every BW-critical report chain and assign one accountable owner to each chain. The next step classifies interfaces, extractors, mappings, master-data dependencies, and historical-comparison risks before transport planning dominates the schedule. The team then defines joint review rituals in which technical staff and business validators inspect logic changes through the same artifact trail. Offshore scaling becomes safer only after these controls are in place. Without this sequence, the project expands delivery capacity before it has created a stable language for coordination. This interpretation supports a qualified preference for centralized or tightly governed hybrid models. The preference does not rest on an assumption that offshore work lacks quality. It rests on a narrower claim about BW delivery. Instability grows when distributed structures divide execution from meaning and when the organization discovers that split only through disputed reports. Once that pattern begins, correction work grows faster than planned savings. The expected efficiency of fragmentation can be offset by rework, delayed sign-off, and loss of trust.

The study has several limitations. First, it is based on conceptual synthesis and prior literature rather than direct empirical measurement of SAP BW projects. Second, the findings apply primarily to BW-heavy S/4HANA

programs with legacy reports, custom transformations, and strong dependence on historical comparability. Third, the proposed monitoring metrics are analytical recommendations, not tested quantitative indicators. Future research should verify the model through project-level data, including defect reopening rates, validation latency, post-go-live report disputes, and ownership patterns across offshore and onshore teams.

5. Conclusion

The article suggests that BW instability in S/4HANA-related programs tends to increase where migration pressure intersects with organizational fragmentation. Reporting defects, reconciliation delays, and trust loss tend to emerge more readily when the project divides ownership of extraction, transformation, validation, and business interpretation across separate teams without a durable cross-layer control point.

The main contribution of this study lies in demonstrating that organizational team structure can be treated as a primary explanatory variable for reporting stability in SAP BW environments. This result is unique because it connects delivery model design with concrete elements of the reporting chain, including extractors, transformations, reconciliation logic, and post-go-live validation, enabling a more precise diagnosis of reporting risks than existing approaches focused on technical migration alone. The study introduces a novel conceptual model based on three forms of continuity—interpretation, validation, and ownership—which together explain the resilience of BW reporting systems under transformation conditions. This framework extends beyond existing literature by integrating organizational, technical, and validation dimensions into a single analytical structure.

The author discovered that reporting instability in SAP BW environments during S/4HANA-related transformation is not only a technical consequence of migration complexity, data-quality defects, or insufficient testing, but also an organizational effect produced by fragmented responsibility across extraction, transformation, reconciliation, and business validation layers. The uniqueness of this finding lies in treating team structure as a direct source of reporting risk, rather than as a secondary project-management condition. The author's contribution consists in converting SAP BW/BI project experience into a practical conceptual model based on interpretive continuity, validation continuity, and ownership continuity. This model explains why centralized or tightly governed hybrid delivery structures are more capable of preserving reporting reproducibility than fragmented offshore and onshore arrangements. The literature supports the proposed hypothesis that SAP BW delivery stability is more likely when semantic and architectural control functions remain organizationally concentrated. The novelty of the findings is confirmed by the consistent identification of coordination-induced reporting risks across multiple sources, which elevates the study beyond a routine synthesis of known factors.

Acknowledgements

These and the Reference headings are in bold. The text below continues as normal.

References

- [1] Holland, R. (2024, March). SAP S/4HANA migration: Benchmark report. SAPinsider. <https://sapinsider.org/wp-content/uploads/2025/04/SAPinsider-2024-03-SAP-S4HANA-Migration->

Report.pdf

- [2] Chang, Y.-W., & Chen, J. (2025). An empirical investigation of critical success factors in implementing digital transformation. *Technological Forecasting and Social Change*, 217, 124161. <https://doi.org/10.1016/j.techfore.2025.124161>

- [3] Chang, Y. (2025). Analyzing the impact of ERP and BI integration on business performance: The technology-organization-environment framework and balanced scorecard perspective. *Journal of Global Information Management (JGIM)*, 33(1), 1–25. <https://doi.org/10.4018/JGIM.389254>

- [4] Bernardo, B. M. V., Mamede, H. S., Barroso, J. M. P., & Santos, V. M. P. D. (2024). Data governance & quality management—Innovation and breakthroughs across different fields. *Journal of Innovation & Knowledge*, 9(4), 100598. <https://doi.org/10.1016/j.jik.2024.100598>

- [5] Acev, D., Biyani, S., Rieder, F., et al. (2025). Systematic analysis of data governance frameworks and their relevance to data trusts. *Management Review Quarterly*. <https://doi.org/10.1007/s11301-025-00545-1>

- [6] Butarbutar, Z. T., Handayani, P. W., Suryono, R. R., & Wibowo, W. S. (2023). Systematic literature review of critical success factors on enterprise resource planning post implementation. *Cogent Business & Management*, 10(3). <https://doi.org/10.1080/23311975.2023.2264001>

- [7] Andrade, A. S. L. de, Jackson, V., Prikladnicki, R., & van der Hoek, A. (2024). On meetings involving remote software teams: A systematic literature review. *Information and Software Technology*, 175, 107541. <https://doi.org/10.1016/j.infsof.2024.107541>

- [8] Alzoubi, Y. I., & Mishra, A. (2024). Enterprise architecture contribution in distributed agile software development. *Systems Engineering*, 27, 570–584. <https://doi.org/10.1002/sys.21739>

- [9] Nakayama, M., Hustad, E., Sutcliffe, N., & Beckfield, M. (2024). Organic transformation of ERP documentation practices: Moving from archival records to dialogue-based, agile throwaway documents. *International Journal of Information Management*, 74, 102717. <https://doi.org/10.1016/j.ijinfomgt.2023.102717>