

Data Conversion Frameworks for PeopleSoft-to-Workday Migration: Handling Payroll, Benefits, and Absence Domains

Ujwal Dyasani

Lead Software Engineer Integrations, USA

ABSTRACT: Migrations from PeopleSoft Human Capital Management to Workday represent one of the most data-intensive classes of enterprise system transition, requiring the conversion of deeply configured, tenant-specific data structures accumulated over many years of PeopleSoft implementation into Workday's business-object-oriented data model. Among the functional domains involved in such a migration, Payroll, Benefits, and Absence data present disproportionate conversion complexity relative to core worker and organizational data, owing to their dependence on historical balances, plan-year-scoped configuration, and calculation-sensitive data that must reconcile precisely against legacy system output.

This research article presents a structured data conversion framework for PeopleSoft-to-Workday migrations, developed and refined across multiple enterprise implementation engagements, that decomposes the conversion effort into assessment, extraction, mapping, transformation, validation, load, and reconciliation phases, applied consistently across the Payroll, Benefits, and Absence domains. The article documents domain-specific conversion object inventories, representative field-mapping structures, data volume and complexity characteristics, and a parallel-payroll validation methodology used to confirm conversion accuracy prior to cutover.

Findings are presented alongside data tables summarizing conversion object counts, complexity ratings, and testing cycle timelines drawn from generalized case observations. The article concludes with a discussion of study limitations, directions for continued research into automated conversion validation, and practical governance recommendations for migration programs undertaking similar conversions.

KEYWORDS: PeopleSoft, Workday, data conversion, HCM migration, payroll conversion, benefits conversion, absence management, ETL, data reconciliation, parallel payroll testing.

FRAMING STATEMENT

The central premise of this research is that Payroll, Benefits, and Absence conversion cannot be treated as a generic data-migration exercise; each domain carries distinct calculation dependencies and historical-data requirements that demand domain-specific conversion logic, dedicated reconciliation criteria, and domain-aware testing cycles layered on top of a common conversion framework.

I. INTRODUCTION

Enterprise adoption of Workday as a cloud Human Capital Management platform has driven a sustained wave of migrations away from on-premises PeopleSoft HCM installations, many of which had been configured, extended, and customized over one to two decades of continuous use. While such migrations touch every functional area of workforce data, the Payroll, Benefits, and Absence domains consistently emerge as the most conversion-intensive, because each domain carries calculation-sensitive historical data whose accuracy directly affects employee pay, benefit eligibility, and leave balances from the moment of cutover forward.

Unlike core worker and position data, which can often be converted with a relatively direct source-to-target field mapping, Payroll, Benefits, and Absence data typically require the reconstruction of derived balances, the reinterpretation of plan-year-scoped configuration into Workday's equivalent constructs, and rigorous reconciliation against legacy system output before an organization can safely rely on the converted data for live payroll processing. Underestimating this complexity is a recurring source of migration schedule and cost overrun observed across enterprise HCM transition programs.

1.1 Motivation

This research is motivated by the absence of a widely documented, domain-specific conversion methodology for the Payroll, Benefits, and Absence areas of a PeopleSoft-to-Workday migration, in contrast to the comparatively well-documented conversion patterns for core worker, position, and organizational data. Migration teams frequently reconstruct domain-specific conversion logic independently on each engagement, without a shared reference framework, resulting in inconsistent conversion quality and repeated rediscovery of the same domain-specific pitfalls.

1.2 Research Objectives

- **Define a domain-agnostic conversion framework** spanning assessment, extraction, mapping, transformation, validation, load, and reconciliation phases, applicable across Payroll, Benefits, and Absence conversions.
- **Document domain-specific conversion object inventories** and representative field-mapping structures for each of the three domains, grounded in PeopleSoft source structures and Workday target business objects.
- **Characterize relative conversion complexity** across domains using consistent criteria, to support migration planning and effort estimation.
- **Present a parallel-payroll validation methodology** for confirming conversion accuracy prior to cutover, with emphasis on reconciliation criteria specific to each domain.
- **Provide governance recommendations** for sustaining data conversion quality across the full migration lifecycle, from initial assessment through post-cutover stabilization.

1.3 Scope and Audience

This article addresses data conversion specifically within the Payroll, Benefits, and Absence domains of a PeopleSoft-to-Workday migration; it does not provide a comprehensive treatment of core worker/position/organizational data conversion, integration architecture, or organizational change management, except where those areas directly intersect with domain data conversion. The intended audience includes data migration architects, functional conversion leads, and technical program managers responsible for planning or executing a PeopleSoft-to-Workday migration.

II. BACKGROUND: PEOPLESOFT AND WORKDAY DATA ARCHITECTURES

2.1 Structural Differences Between the Two Platforms

PeopleSoft HCM organizes data around a relational schema of discrete, often highly normalized tables, frequently extended over time with customer-specific fields, effective-dated rows, and custom tables layered alongside delivered PeopleSoft structures. Workday, by contrast, organizes data around a business-object model in which related data (for example, a worker's compensation, benefit elections, and absence balances) is associated with a governed business object and exposed through calculated fields, business processes, and web services rather than direct table access.

This architectural difference means that data conversion is rarely a mechanical field-to-field mapping exercise; converted data must be reshaped to fit Workday's object model, including resolution of effective-dated PeopleSoft history into the discrete point-in-time or as-of-date structures Workday expects, and reconstruction of derived values that PeopleSoft may have stored explicitly but that Workday instead calculates dynamically.

2.2 Comparative Architectural Summary

Table.1.Comparative architectural characteristics relevant to data conversion planning

Characteristic	PeopleSoft HCM	Workday HCM
Data model	Relational, effective-dated tables	Business-object model with calculated fields
Customization pattern	Custom tables, PeopleCode, panels	Configurable business processes, custom fields
History representation	Explicit effective-dated rows	As-of-date business object versions
Payroll calculation storage	Stored calculated results in paysheet tables	Calculated at runtime from configuration and inputs
Integration pattern	Batch/Application Engine, direct SQL	Web services, EIBs, Workday Studio, report-based extracts

2.3 Why Payroll, Benefits, and Absence Are Disproportionately Complex

- Payroll conversion must reconcile historical earnings, deductions, and tax balances to the cent against legacy system output, since any discrepancy directly affects employee pay and year-end tax reporting.
- Benefits conversion must correctly interpret plan-year-scoped elections, dependent and beneficiary relationships, and evidence-of-insurability status that may span multiple PeopleSoft benefit program configurations.
- Absence conversion must reconstruct accrual balances, carryover rules, and negative balance history in a manner consistent with Workday's absence plan configuration, which may not mirror the PeopleSoft absence system's accrual logic exactly.

III. PROBLEM STATEMENT: CONVERSION COMPLEXITY ACROSS DOMAINS

The central problem addressed by this research is the disproportionate share of migration risk, effort, and rework concentrated in the Payroll, Benefits, and Absence conversion workstreams relative to their proportion of total data volume, and the corresponding need for a domain-aware conversion methodology that anticipates this complexity rather than discovering it during execution.

3.1 Recurring Complexity Drivers

- **Historical balance reconstruction.** Both Payroll and Absence domains require converting cumulative balances (year-to-date earnings, accrual balances) that must reconcile to legacy system totals at a specific point in time, rather than simply transferring current-state records.
- **Configuration reinterpretation.** Benefits and Absence domains require translating PeopleSoft plan and program configuration into functionally equivalent Workday configuration before the underlying election or balance data can be converted meaningfully.
- **Calculation sensitivity.** Payroll data in particular is sensitive to exact calculation logic; a converted balance that is directionally correct but numerically imprecise is insufficient, since payroll processing requires exact reconciliation.
- **Data quality inheritance.** Data quality issues accumulated in PeopleSoft over many years (duplicate dependent records, inconsistent effective-dating, orphaned records) surface during conversion and frequently require remediation before conversion can proceed.

3.2 Consequences of Underestimating Domain Complexity

Migration programs that apply a uniform conversion approach across all domains without accounting for the specific complexity drivers above frequently encounter schedule slippage concentrated in the Payroll, Benefits, and Absence workstreams, discovered during parallel testing cycles rather than during initial planning. In several engagements reviewed for this research, Payroll and Benefits conversion rework accounted for a disproportionate share of total conversion remediation effort relative to their share of total converted data volume.

IV. RESEARCH METHODOLOGY

This research synthesizes observations from multiple enterprise PeopleSoft-to-Workday migration engagements, combined with a review of published Workday and PeopleSoft technical documentation and established data migration methodology literature.

4.1 Approach

1. Conversion object inventory: For each domain, catalogue the PeopleSoft source tables and Workday target business objects involved in conversion, and classify each conversion object by complexity driver.
2. Field-mapping structure review: Document representative field-mapping patterns between PeopleSoft source fields and Workday target fields, including transformation logic required to bridge structural differences.
3. Complexity and volume characterization: Assess relative conversion complexity and typical data volume for each conversion object, informing the effort estimation guidance in Section 6 through Section 8.
4. Validation methodology synthesis: Document the parallel-payroll and domain-specific reconciliation testing approaches applied across reviewed engagements.
5. Governance synthesis: Consolidate recurring governance practices that sustained conversion quality across the full migration lifecycle.

4.2 Validity Considerations

Conversion object names, complexity ratings, and volume figures presented in this article are generalized to reflect commonly observed patterns across multiple engagements rather than a single named implementation, and specific tenant configuration details have been abstracted to protect confidentiality. Figures should be interpreted as representative planning guidance rather than universal constants applicable to every migration.

V. DATA CONVERSION FRAMEWORK OVERVIEW

The framework developed through this research organizes conversion work into seven sequential, domain-agnostic phases, applied independently but consistently within each of the Payroll, Benefits, and Absence domains.

5.1 Phase Definitions

- **Assessment.** Inventory source data objects, assess data quality, and confirm target Workday configuration is sufficiently defined to receive converted data.
- **Extraction.** Extract source data from PeopleSoft using queries, Application Engine processes, or delivered extract utilities, preserving effective-dated history where required.
- **Mapping.** Define field-level and structural mappings from PeopleSoft source structures to Workday target business objects, including transformation and lookup logic.
- **Transformation.** Execute the mapping logic, typically through a dedicated ETL tool or scripted transformation layer, producing Workday-compatible load files.
- **Validation.** Validate transformed data against business rules and Workday configuration constraints prior to load, catching errors before they reach the target tenant.
- **Load.** Load validated data into the Workday tenant using Workday's supported load mechanisms (Enterprise Interface Builder templates, web services, or the Workday Studio runtime).
- **Reconciliation.** Reconcile loaded data in Workday against source system totals and expected outcomes, with domain-specific reconciliation criteria described in Sections 6 through 8.

5.2 Iterative Application Across Mock Conversion Cycles

In every engagement reviewed for this research, the seven-phase framework was applied across multiple mock conversion cycles rather than executed once, with each successive cycle incorporating remediation of defects identified during the validation and reconciliation phases of the prior cycle. This iterative structure is essential given the calculation sensitivity of the Payroll and Absence domains in particular, where conversion logic defects are rarely fully apparent until reconciliation against real historical data is attempted.

MOCK CONVERSION CYCLE

A complete execution of the extraction-through-reconciliation phases against a representative data set, conducted prior to the final production conversion, used to surface and remediate conversion defects under controlled conditions.

VI. PAYROLL DOMAIN CONVERSION

Payroll conversion is typically the most calculation-sensitive and reconciliation-intensive domain in a PeopleSoft-to-Workday migration, owing to the requirement that converted historical earnings, deductions, and tax balances reconcile precisely against PeopleSoft payroll register and year-end tax reporting output.

6.1 Principal Conversion Objects

Table.2. Representative Payroll domain conversion objects, source-to-target mapping, and relative complexity

Conversion Object	PeopleSoft Source	Workday Target	Complexity
Year-to-date earnings balances	PAY_EARNINGS, PAY_CHECK	Payroll Results History	High
Deduction balances	PAY_DEDUCTION	Benefit/Deduction Balances	High
Tax balances	PAY_TAX_BAL, PAY_TAX_LINE	Tax Balance object	High
Garnishment history	GARN_SPEC, GARN_DED_TBL	Garnishment Case	Medium
Direct deposit elections	DIRECT_DEPOSIT	Payment Elections	Low
Employee tax elections	PAY_US_TAX_DATA	Withholding Elections	Medium

6.2 Balance Reconstruction Approach

Because Workday does not simply replicate PeopleSoft's stored balance records, converted year-to-date balances must typically be reconstructed as of a defined conversion as-of-date by summarizing detailed PeopleSoft earnings and deduction history, then loaded into Workday's payroll results history structures in a form Workday's own calculation engine can build upon going forward. This reconstruction step is the most common source of reconciliation discrepancy identified during mock conversion cycles.

6.3 Reconciliation Criteria

- Converted year-to-date gross, net, and tax balances must match PeopleSoft payroll register totals exactly, to the cent, as of the defined conversion as-of-date.
- Converted deduction balances must reconcile against PeopleSoft deduction history reports for each active deduction code.
- The first post-cutover payroll run in Workday must produce results consistent with parallel payroll testing performed against the same population in the legacy system.

VII. BENEFITS DOMAIN CONVERSION

Benefits conversion centers on accurately translating PeopleSoft benefit program and plan configuration into Workday benefit plan configuration, then converting individual employee elections, dependent and beneficiary data, and evidence-of-insurability status consistent with that configuration.

7.1 Principal Conversion Objects

Table.3. Representative Benefits domain conversion objects, source-to-target mapping, and relative complexity.

Conversion Object	PeopleSoft Source	Workday Target	Complexity
Benefit elections	BEN_PROG_PARTIC, BEN_ELECTION	Benefit Enrollment Event	High
Dependent/beneficiary data	DEPENDENT_BENEF	Dependent, Beneficiary objects	Medium
Life event history	BAS_ACTIVITY	Benefit Change Event	Medium
Evidence of insurability status	EOI_REQUEST	EOI Tracking	Medium
FSA/HSA contribution elections	FSA_ELECTION	Flexible Spending Election	Medium
COBRA participant data	COBRA_ACTIVITY	COBRA Event	Low

7.2 Plan-Year Configuration Dependency

Because benefit elections are meaningful only in the context of the plan configuration under which they were made, benefits conversion cannot proceed independently of benefits configuration; Workday benefit plans, coverage tiers, and rate structures for the relevant plan years must be configured and validated before election data conversion is attempted, making benefits conversion sequencing a common dependency bottleneck in overall migration planning.

7.3 Reconciliation Criteria

- Converted elections must match the employee's current active election as reflected in the PeopleSoft benefits system as of the conversion as-of-date, including coverage tier and dependent coverage.
- Dependent and beneficiary relationships must resolve correctly to the corresponding worker record with no orphaned or duplicated dependent entries.
- In-progress life events and pending evidence-of-insurability cases must convert with their correct pending status rather than being incorrectly finalized or dropped.

VIII. ABSENCE DOMAIN CONVERSION

Absence conversion requires reconstructing accrual balances and carryover history in a manner consistent with Workday's absence plan and accrual configuration, which may compute accruals differently than the source PeopleSoft Absence Management or Time and Labor configuration.

8.1 Principal Conversion Objects

Table.4. Representative Absence domain conversion objects, source-to-target mapping, and relative complexity

Conversion Object	PeopleSoft Source	Workday Target	Complexity
Accrual balances	ABS_BAL_TBL	Absence Plan Balance	High
Carryover history	ABS_ENTITLE_BAL	Absence Plan Balance (carryover)	Medium
Pending/approved absence requests	ABS_EVENT	Time Off / Leave Request	Medium
Leave-of-absence case data	LOA_DATA	Leave of Absence Case	Medium
Negative balance history	ABS_BAL_TBL (negative rows)	Absence Plan Balance (negative)	High

8.2 Accrual Logic Reconciliation

A recurring challenge identified across reviewed engagements is that Workday's absence accrual calculation logic, once configured, will recompute future accruals going forward from the converted opening balance; if the converted opening balance does not correctly reflect the net effect of PeopleSoft's accrual, usage, and carryover rules as of the conversion as-of-date, subsequent Workday-calculated accruals will diverge from what employees and managers expect based on legacy system history.

8.3 Reconciliation Criteria

- Converted opening accrual balances must match PeopleSoft absence balance reports as of the conversion as-of-date for every active absence plan and employee combination.
- Carryover amounts must reflect the correct application of PeopleSoft carryover caps and forfeiture rules as of the plan year boundary nearest the conversion date.
- Any negative balance positions present in the legacy system must convert with correct sign and magnitude rather than being floored to zero.

IX. CROSS-DOMAIN DATA QUALITY AND RECONCILIATION

While Sections 6 through 8 address domain-specific conversion objects, several data quality and reconciliation concerns recur across all three domains and warrant coordinated treatment.

9.1 Cross-Domain Data Volume and Complexity Summary

Table.5. Relative conversion object count, data volume, and complexity across domains, shown against core worker data as a reference point

Domain	Typical Conversion Object Count	Relative Data Volume	Overall Rating	Complexity
Payroll	6–10 core objects	Very High (multi-year history)	High	
Benefits	6–9 core objects	High (multi-plan-year elections)	Medium-High	
Absence	5–7 core objects	Medium (balance + transaction history)	High	
Core Worker/Position (reference)	10–14 core objects	High (full population, single state)	Medium	

9.2 Common Data Quality Issues Surfaced During Conversion

- Duplicate or conflicting effective-dated rows in PeopleSoft source tables, often resulting from historical data correction activity that did not fully supersede prior rows.
- Orphaned dependent, beneficiary, or garnishment records referencing worker identifiers no longer present in the active population extract.
- Inconsistent use of custom fields or panels across business units, resulting in conversion logic that must branch by organizational unit rather than applying uniformly.
- Historical records predating a defined data retention or conversion cutoff, requiring an explicit organizational decision on how far back converted history must extend.

9.3 Coordinated Reconciliation Sequencing

Because Payroll conversion accuracy depends in part on correctly converted Benefits deduction elections and Absence-driven pay impacts, the framework recommends sequencing domain reconciliation so that Benefits and Absence conversion validation completes, at least for the population and plan year in scope, before final Payroll balance reconciliation is attempted, reducing the risk of diagnosing a Payroll discrepancy that actually originates in an upstream Benefits or Absence conversion defect.

X. TESTING STRATEGY AND PARALLEL PAYROLL VALIDATION

Validation of converted Payroll, Benefits, and Absence data culminates in parallel testing, in which live production processing is run concurrently in both the legacy PeopleSoft environment and the target Workday tenant, with results compared to confirm conversion accuracy before cutover.

10.1 Typical Testing Cycle Structure

Table.6. Representative testing cycle structure culminating in parallel payroll validation prior to cutover

Cycle	Primary Focus	Typical Duration	Exit Criteria
Mock Conversion 1	Initial object-level load and structural validation	2–3 weeks	All objects load without structural error
Mock Conversion 2	Field-level and balance-level accuracy validation	3–4 weeks	Balance reconciliation within defined tolerance
Mock Conversion 3 / Parallel Payroll 1	Full reconciliation and first parallel payroll run	2 pay cycles	Net pay variance within zero-tolerance threshold
Parallel Payroll 2	Confirmation run addressing Parallel 1 findings	1–2 pay cycles	Zero unexplained variances
Final Conversion (Cutover)	Production data conversion for go-live	3–5 days	Reconciliation sign-off by domain owners

10.2 Parallel Payroll Validation Approach

Parallel payroll validation runs the same input population and time period through both the legacy PeopleSoft payroll process and the converted Workday tenant, comparing gross pay, each deduction and tax line, and net pay at the individual employee level. Because this comparison exercises not only converted historical balances but also the ongoing calculation configuration built in Workday, it serves as the definitive validation gate for both data conversion accuracy and payroll configuration correctness simultaneously.

10.3 Domain-Specific Testing Emphasis

- **Payroll testing** emphasizes exact balance and calculation-result reconciliation across the full parallel period, with zero tolerance for unexplained variance.
- **Benefits testing** emphasizes correct coverage tier, dependent association, and rate application for a representative sample spanning every distinct benefit plan and eligibility scenario.
- **Absence testing** emphasizes correct opening balance and at least one full accrual cycle post-conversion to confirm ongoing Workday-calculated accruals track correctly from the converted opening position.

XI. GOVERNANCE AND CUTOVER PLANNING

11.1 Domain Ownership Model

Sustained conversion quality across Payroll, Benefits, and Absence requires named functional owners for each domain who hold accountability for reconciliation sign-off, distinct from the technical conversion development team, ensuring that reconciliation results are reviewed against business expectations rather than only against technical load-success criteria.

11.2 Cutover Sequencing Considerations

- Establish a single, clearly communicated conversion as-of-date across all three domains to avoid balance mismatches arising from inconsistent cutoff points between domains.
- Sequence final production conversion activities so that Benefits and Absence data are loaded and validated before final Payroll balance conversion, consistent with the reconciliation sequencing discussed in Section 9.
- Define and rehearse a rollback or contingency procedure in the event that final reconciliation identifies a defect discovered too late to remediate within the planned cutover window.

11.3 Post-Cutover Stabilization

Conversion governance should extend beyond go-live through at least one full payroll cycle and one full benefits plan year transition, since certain conversion defects - particularly in Absence accrual logic and Benefits rate calculation - may not surface until the first live processing cycle exercises the converted configuration under real operating conditions rather than test conditions.

XII. CASE OBSERVATIONS

12.1 Observation A - Payroll Balance Reconstruction Discrepancy

In one reviewed engagement, initial converted year-to-date earnings balances failed to reconcile against PeopleSoft payroll register totals for a subset of the population subject to mid-year pay group transfers. Root cause analysis traced the discrepancy to conversion logic that summarized earnings by current pay group rather than by the correct historical pay group association at the time each earnings record was created. Correcting the summarization logic to respect historical pay group association resolved the discrepancy in the subsequent mock conversion cycle.

12.2 Observation B - Benefits Dependent Data Duplication

A benefits conversion review identified duplicate dependent records for a portion of the population, traced to a historical PeopleSoft data correction process that had created new dependent rows rather than updating existing ones. Introducing a deduplication step in the extraction phase, keyed on dependent name and date of birth in combination with worker identifier, resolved the duplication prior to load.

12.3 Observation C - Absence Carryover Miscalculation

An absence conversion review found that initial converted carryover balances did not correctly apply PeopleSoft's plan-specific carryover cap, resulting in overstated opening balances for employees who had accrued above the cap

threshold. Revising the conversion transformation logic to explicitly apply each plan's carryover cap prior to loading the opening balance resolved the discrepancy.

12.4 Cross-Case Observations

- In each case, the root cause was traced to conversion logic that had not fully accounted for a historical configuration nuance (pay group transfer history, data correction artifacts, plan-specific caps) rather than to a defect in the overall framework structure.
- Every identified discrepancy was surfaced during a mock conversion or parallel testing cycle rather than in production, consistent with the value of the iterative testing structure described in Section 10.
- Resolution in each case required close collaboration between technical conversion developers and functional domain owners with detailed knowledge of legacy configuration history.

XIII. LIMITATIONS OF THE STUDY

- Observations are drawn from a limited set of enterprise PeopleSoft-to-Workday migration engagements and may not represent the full diversity of PeopleSoft configuration encountered across all customer environments.
- Conversion object inventories and complexity ratings presented in Sections 6 through 8 are representative rather than exhaustive; specific tenant configurations may introduce additional conversion objects not covered in this article.
- This research addresses Payroll, Benefits, and Absence domain conversion specifically and does not provide comprehensive treatment of core worker, compensation, or organizational data conversion, which involve a different complexity profile.
- Testing cycle durations presented in Section 10 reflect representative planning ranges and will vary based on population size, configuration complexity, and organizational testing resource availability.
- Case observations have been generalized and anonymized, which limits independent verification of specific quantitative findings.

XIV. FUTURE RESEARCH DIRECTIONS

- **Automated reconciliation tooling.** Development of tooling that automatically compares converted Workday balances against PeopleSoft source system totals across full populations, reducing reliance on manual sample-based reconciliation.
- **Standardized conversion object taxonomies.** A shared, versioned taxonomy of PeopleSoft-to-Workday conversion objects across all HCM domains, maintained as a community reference rather than rebuilt independently on each engagement.
- **Predictive complexity estimation models.** Further research into estimation models that predict conversion effort and defect likelihood from measurable PeopleSoft configuration characteristics, such as customization depth and historical data volume.
- **Extension to other source ERP/HCM platforms.** This research focused specifically on PeopleSoft source systems; extending the framework's domain-specific reconciliation approach to other legacy HCM platforms migrating to Workday would broaden its applicability.

XV. CONCLUSION

Payroll, Benefits, and Absence conversion consistently present the greatest data conversion complexity in PeopleSoft-to-Workday migrations, driven by historical balance reconstruction requirements, plan-year-scoped configuration dependencies, and exacting calculation-sensitivity that demands precise reconciliation rather than approximate correctness. The seven-phase conversion framework and domain-specific conversion object inventories presented in this article provide migration teams with a structured, repeatable approach to planning and executing this work, grounded in patterns observed consistently across multiple enterprise engagements.

Equally important to the technical conversion logic is the iterative testing structure described in Section 10, culminating in parallel payroll validation, and the governance practices described in Section 11, which together ensure that conversion defects are surfaced and remediated during controlled testing cycles rather than discovered in live production processing after cutover.



As enterprise adoption of Workday continues and the population of organizations undertaking PeopleSoft-to-Workday migrations grows, the domain-specific conversion framework and reconciliation criteria presented in this article offer a practical reference for migration teams seeking to reduce the schedule and cost risk historically concentrated in the Payroll, Benefits, and Absence conversion workstreams.

REFERENCES

The following references reflect publicly available technical documentation, data migration methodology literature, and industry guidance current as of the publication date of this article (September 2020).

1. Oracle Corporation. (2019). PeopleSoft HCM 9.2: Application Fundamentals. Oracle PeopleSoft Online Help Documentation.
2. Oracle Corporation. (2019). PeopleSoft Payroll for North America 9.2: PeopleBook. Oracle PeopleSoft Online Help Documentation.
3. Oracle Corporation. (2019). PeopleSoft Benefits Administration 9.2: PeopleBook. Oracle PeopleSoft Online Help Documentation.
4. Oracle Corporation. (2019). PeopleSoft Absence Management 9.2: PeopleBook. Oracle PeopleSoft Online Help Documentation.
5. Workday, Inc. (2020). Workday HCM: Core Compensation and Payroll Reference Guide. Workday Community Documentation, accessed prior to September 2020.
6. Workday, Inc. (2020). Workday Benefits Administration Reference Guide. Workday Community Documentation, accessed prior to September 2020.
7. Workday, Inc. (2020). Workday Absence Management Reference Guide. Workday Community Documentation, accessed prior to September 2020.
8. Workday, Inc. (2020). Workday Data Conversion Methodology and Enterprise Interface Builder (EIB) Reference. Workday Community Documentation, accessed prior to September 2020.
9. Kimball, R., & Caserta, J. (2004). The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data. Wiley Publishing.
10. Kimball, R., & Ross, M. (2013). The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling (3rd ed.). Wiley Publishing.
11. DAMA International. (2017). DAMA-DMBOK: Data Management Body of Knowledge (2nd ed.). Technics Publications.
12. Loshin, D. (2010). The Practitioner's Guide to Data Quality Improvement. Morgan Kaufmann/Elsevier.
13. Redman, T. C. (2008). Data Driven: Profiting from Your Most Important Business Asset. Harvard Business Press.
14. [Project Management Institute. (2017). A Guide to the Project Management Body of Knowledge (PMBOK Guide) (6th ed.). Project Management Institute.
15. International Institute of Business Analysis. (2015). A Guide to the Business Analysis Body of Knowledge (BABOK Guide) (v3). International Institute of Business Analysis.
16. Inmon, W. H., & Linstedt, D. (2014). Data Architecture: A Primer for the Data Scientist. Morgan Kaufmann/Elsevier.