International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

Solution-Led Integration Architecture in Oracle EBS: A Dual Case Study from Foundational Enterprise Engagements

Sreedhar Chenna

Principal Consultant, IBM, USA

ABSTRACT: In this article, you will learn about two of the most high-profile enterprise implementations to have produced a comprehensive catalogue of skills regarding Oracle EBS Integration Architecture (EBSIA). The first implementation involved a large telecommunications company in Asia and delivered a solution-led integrated design through the use of Oracle Application Integration Architecture (AIA) design tools and OAGIS Business Object Document (BOD). Governed by BODs, this design delivered the infrastructure necessary to enable the two existing systems to be integrated without any modification to either system's existing technical infrastructure, as well as the ability for existing users to access both systems independently. The author served as the solution lead on this project and was responsible for all aspects of the technical design, implementation of a proof-of-concept, and delivery of the solution (which achieved zero-touch order automation via the sales order process). The second project was with a global technology company and involved the technical configuration of the Oracle EBS financial applications to support their business operations. The key deliverables for this enterprise project included the AUTOLOCKBOX interface to receive payment data from banks, migrating the data from Accounts Payable to other financial modules, and creating custom reports that supported the Order to Cash process. Both enterprise projects reinforced the need for organizations to employ ERP expertise in two distinct areas: strategic architectural design implementations and technical execution of the architectural designs using an Oracle AIA and AIM hybrid method. The results achieved provided the author with valuable lessons for his future career opportunities and will assist organizations with managing their talent pool of ERP implementation professionals.

KEYWORDS: Oracle E-Business Suite (EBS), Oracle Application Integration Architecture (AIA), OAGIS BOD, zero-touch automation, AUTOLOCKBOX, Payables data migration, AIM methodology, integration architecture.

I. INTRODUCTION

1.1 Background and Context

Large companies in many industries rely on Enterprise Resource Planning (ERP) systems to run their business operations. Oracle E-Business Suite (EBS) has the largest share in the ERP segment of the Fortune 500 (in particular industries like telecommunications, manufacturing, financial services, and technology).

The business value of ERP implementations does not come from the functional capabilities of individual ERP systems; rather, value arises from how well ERP integrates with the organization's internal modules, such as order management with inventory and receivables with the general ledger, and how well ERP integrates with the organization's external trading partners, such as customer CRM systems, supplier portals, and bank lockbox networks.

ERP integration can be difficult Oracle EBS, especially R12.1.3 versions, creates a complicated technical landscape. There are many modules with their own unique data model, business processes that operate on a workflow basis, concurrent program architecture, and interface tables that must be filled with high accuracy. Additionally, an organization's enterprise environment typically requires real-time, or close to real-time, integration; graceful exception handling, auditability, and high-performance, scalable solutions.

This article describes a significant five-year period in their career that shaped the way they would work with major clients, including Schneider Electric and Paychex. This article is not intended to be biographical instead, it serves as a model for skill development, particularly in mastering the assembly and implementation (or orchestration) of the different layers of the technology integration module. Two separate examples will be reviewed: one for a telecommunications company concentrating on creating an integrated architecture and client acceptance and one for a

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

global enterprise technology company concentrating on developing interfaces and migrating legacy data. Each example exemplified different roles in various frameworks and modules and provided lessons on the various operational impacts from order-to-cash processes to accounting to migration.

This article outlines the six separate sections. Literature Survey (Section 2) reviews the related literature within five areas of research: canonical data models and OAGIS BODs; zero-touch automation, cash application, and AUTOLOCKBOX; migrating legacy data; and methods to implement, including testing and warranty processes. Methodology (Section 3) describes the five-phase unified approach used in both projects.Section 4 presents a breakdown of the governance structure, the roles, and the input and output of each phase, as well as diagrams for both the zero-touch sales order and the AUTOLOCKBOX interfaces, metrics, a table of results, and visual charts in the Results and Architecture section. Section 5 contains a summary of four key findings, identifies limitations, and discusses the implications of these findings for both practice and academia. Section 6 provides a summary of the contributions to this article and outlines future research possibilities.

II. LITERATURE SURVEY

This literature review synthesizes research findings from multiple sources to create one comprehensive literature review, including a review of 50+ peer-reviewed journals, practitioners' publications, and official documents published by Oracle. These reviewed documents were then categorized into five thematic areas that can be used in future engagements.

2.1 Canonical Data Models and OAGIS BODs

"Semiotic mismatches" occur when different parties refer to the same business issue but have different models around it. To solve these issues, Open Applications Group Integration Specification (OAGIS) Business Object Documents (BODs) provide a canonical data model.

There are two main areas to a business object document: application area and data area, according to the OAGIS BOD architecture specification. The application area includes [1]:

- CreationDateTime - (A date/time stamp). When the BOD was created. Will not change through the BOD life cycle.
- BODID - Global Unique Identifier (GUID). The GUID identifies each BOD individually from all other BODs. The legal basis for carrying out binding transactions keeps a record of all transactions, has a way to handle exceptions, tracks BODs that were sent or re-sent successfully, allows for audits, maintains a chain of BOD confirmations, and ensures effective supervision and security.
- The Authorization Identifier describes how the BOD has been BODed into a system and provides for the routing mechanism to return "through the same" to the process that originally inserted it into a receiving system.

The DataArea contains both a verb (the client will perform this action, i.e., PROCESS, CANCEL, ADD, SYNCHRONIZE) and one or more nouns (the object(s) to which the verb is being run against, i.e., PurchaseOrder, SalesOrder, Invoice). It should be noted that OAGIS provides means for extending the number of nouns (items) that can exist through the use of inline extensions.

In the telecommunications integration project described in this article, the SalesOrder BOD was used as the primary integration mechanism. The structured format of the BODs allowed for the completely automated, no-touch processing/fulfillment of all of the data related to the order. All order data was validated at the time of the initial ingestion.

2.2 AIM Methodology and Oracle Implementation Frameworks

Oracle's AIM (Application Implementation Method) [10] is a structured approach that provides an implementation framework for Oracle Applications. AIM encompasses all project aspects, including project planning, as well as defining requirements, modeling business processes, customizing and converting data, and managing organizational change and security. AIM's (Application Implementation Method) seven-phase implementation process consists of definition, operations analysis, solution design, build, business system testing, transition, and production; each phase includes a series of quality control checkpoints. AIM is also integrated with Oracle's Project Management Methodology (PJM), providing oversight for projects. Academic literature provides documentation of AIM's practices and issues that aid in understanding AIM's implementation process, such as that of RUFINO, R. (2004) [4]. Successful implementations using AIM highlighted the value of AIM's structured process and especially how AIM's structured

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

process facilitated the successful implementation of many disparate modules across a complex enterprise technology environment.

2.3 Zero-Touch Automation in Order-to-Cash Cycles

With minimal supervision and very little manual intervention from finance teams, zero-touch automation drastically alters the order-to-cash cycle by incorporating it into daily business operations. Traditional accounts receivable (AR) processes typically overlook other costs, including opportunity costs associated with capital being tied up in working capital, high Days Sales Outstanding (DSOs), and inefficient utilization of the finance team's time. Studies indicate that manual processes, rather than other factors, primarily cause most delays and errors in the O2C process. The operational framework behind zero-touch automation encompasses multiple capabilities [9]:

- Near real-time visibility into accounts receivable without burdensome reporting;
- Advanced modeling techniques used to create predictive cash flow forecasts;
- Automated credit approvals based on dynamic thresholds;
- Elimination of manual reconciliation processes;
- Significant reductions in DSO and working capital requirements;
- Redeployment of finance resources towards analytical and more value-creating work.

Modern autonomous order-to-cash (O2C) platforms powered by Agentic AI produce 90–95% touchless processing and integrate with ERP systems such as Oracle, thus improving cash flow and reducing DSO. Industry examples indicate that autonomous order-to-cash platforms outperform both traditional robots and robotic process automation (RPA) at multiple performance metrics, including cash application rates, days sales outstanding reductions, FTE dependency, and decision intelligence capabilities [7].

The zero-touch sales order flow of an O2C telecommunications company eliminated manual touches on sales orders and exceeded autonomous platform benchmarks. The root cause of this achievement was determining whether or not to validate the data at the time of ingestion instead of relying on manual work queues and using a "fail fast" approach.

2.4 AUTOLOCKBOX and Cash Application Automation

The AutoLockbox program from Oracle is necessary to automate cash applications within Oracle Receivables. This program utilizes a banking service to electronically collect customers' payments, along with their remittance details, before processing the incoming files and digitally generating receipts based upon that data. AR_LOCKBOXES_ALL is the primary repository that stores all configuration information related to lockboxes within the AutoLockbox program, including all lockbox master records and configuration details such as the lockbox's unique identifier, routing code, link to batch source, and link to receipt method [6].

The AutoLockbox system also works with several important tables, like AR_BATCHES_ALL, which keeps track of all transmission records; AR_TRANSMISSIONS_ALL, which tracks the processing steps of a transmission; and AR_PAYMENTS_INTERFACE_ALL, which holds this bank data before it is processed into your organization's receivables. Implementing AutoLockbox reduced cash application time by approximately 75% using standard Oracle patterns, while supporting receipts from multiple contracts processed by banks through Web ADI in Oracle EBS Release 12.2 [2].

The payment lifecycle begins when a source function (e.g., Oracle Payables) creates a payment-related document to be paid (e.g., an invoice). The Oracle Payables function groups payment-related documents into a payment request and sends them to Oracle Payments, which then organizes these requests into payments that result in individual checks or electronic deposits using the Build Payments process. The AUTOLOCKBOX interface is built directly to provide automated cash application functionality for these types of payments [11].

2.5 Data Migration and Reconciliation in Oracle Payables

Migrating payables invoice data from Oracle EBS to a Cloud ERP system is a complicated endeavor. Identifying eligible open payables invoices for an accurate migration requires the use of appropriate criteria. Historically, reconciling the Oracle EBS Payables Trial Balance was performed manually [8], which resulted in time delays and created opportunities for human error. However, research published in 2024 revealed that automating this process of reconciliation would greatly enhance the accuracy of the data conversion. The most significant issues to resolve for a successful migration included managing large volumes of data, addressing invoice errors during the cloud interface conversion, and reconciling differences between legacy and target system formats.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

To implement an effective technique for automating reconciliation requires the use of defined staging tables for the extraction, transformation, and validation of data. The framework described in this paper consists of reconciling extracted invoices against those in the Oracle EBS Payables Trial Balance using tables such as XX_AP_CLOUD_INV_HDR_STG and XX_AP_R12_APTB_STG [5]. Automation has produced far-reaching benefits, reducing reconciliation, requiring 50% less time, eliminating human errors, increasing scalability for large data volumes, increasing the accuracy of financial data, and enabling finance teams to become more focused on strategic activities.

During the Payables data migration process, a three-pass iterative validation approach was employed that resulted in an exceptional 99.7% accuracy rate, which is consistent with the overall best practices in the literature that relate to automated reconciliation.

2.6 Order-to-Cash Automation in B2B Environments

In multi-channel B2B order processing solutions, order processing is a financial data management activity that collaborates to process the "order-to-cash" (O2C) business cycle. The O2C process encompasses the aforementioned steps, beginning with the collection of purchase order data from customer portals, which is then processed through standardized data (SKU mapping) into a sales order within the ERP system, continuing through to the final stage of debt reconciliation. Extracting and manually processing these purchase orders can have many risks, such as input delays of hours per order due to the current method of inputting data from a customer portal; data entries that can have discrepancies due to copy/paste errors and incorrect formatting; and hidden costs, such as employee overtime and SLA penalties, that cause delays in collecting revenue [3].

By utilizing Robotic Process Automation (RPA) solutions, businesses can automate the extraction of order data. Automated portal log on enables a user to download purchase orders, use artificial intelligence (AI)/natural language processing (NLP) to extract data from the downloaded purchase orders; use AI/NLP to perform SKU mapping to the appropriate product in the ERP system, and automatically standardize the way the order data will be sent to the ERP system for processing a sales order.

The advanced zero-touch sales order process mentioned earlier will help automate tasks by allowing sales orders to be created and booked without any manual input, as well as extracting order data automatically. Therefore, organizations can work towards complete automation instead of partial automation.

III. METHODOLOGY

3.1 Methodology Overview

The hybrid methodology used for both engagements was made up of three complementary frameworks that provided a basis for the activities performed:

- For integration-heavy types of work (telecommunications engagement), we followed the Oracle AIA methodology, which focuses on using the OAGIS BODs to model processes, develop service discovery capabilities, define orchestration, and complete end-to-end integration testing.
- For module-heavy types of work (enterprise technology engagement), the Application Implementation Methodology (AIM) was used as we focused on business process mapping/implementation, configuration, interface development, data conversion, and reporting.
- The approach for both engagements included agile (iterative) practices, with two-week sprint cycles, the production of deliverables every two weeks, daily stand-ups to resolve issues as they arise, and bi-weekly stakeholder meetings to review the progress of the project.

3.2 Unified Five-Phase Process

The Unified Process consists of five sequential phases for primary integration flows that also contain iterative feedback loops.

1. **Discovery & Proof of Concept (POC):**

a) Analyze data from the source system and the associated business processes.
b) Identify technical methods and create lightweight POCs to test feasibility.
c) Create performance baselines, and document the assumptions and risks associated with them.
d) The POC validation report and the preliminary project plan are the expected outputs of this phase.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

2. **Client Approval:**

a) Present POC results to client stakeholders and review ROI, resource requirements, and timelines.
b) Obtain formal permission from stakeholders, including one business owner and one technical owner.
c) The signed project charter and the approved resource plan will be the expected outputs of this phase.

3. Build & Configure:

a) Work on specific deliverables, such as Sales Agreement Integration and Auto Lockbox Interface, and record the work done as effort measured in person-days.
b) Follow development standards such as version control and performing code reviews.

4. **Test:**

a) Perform all levels of testing (unit, integration, system, user acceptance), and document the output, including test scripts and a defect log created during testing.
b) Categorize defects based on their severity so that no critical defects exist prior to project completion.

5. **Rollout & Warranty:**

a) Conduct the sequence of deployment and perform smoke testing.
b) Track performance during the initial phase of production and provide technical support for implementation issues.
c) During the warranty support period, the triage of all production issues and knowledge transfer to the support team will be provided.
d) No open severity one (1) or two (2) defects must be present at the end of the warranty phase.

The following describes the respective roles and responsibilities of all personnel within two separate, major engagements: telecommunications and enterprise technology.

- **Telecommunications Engagement:** The Solution Lead is responsible for authoring the POC execution, obtaining client sign-off, leading the internal team, and overseeing the entire project delivery process. The Construction Unit will comprise three to five workers with supervisors from the Construction Section. The project manager assigned to each contract will provide overall supervision of the contracted effort, including managing the construction activities/issues with the construction unit and offering proof-of-concept assistance as required. The Construction Unit Project Manager will assign the resources needed for the construction work and will escalate issues related to construction work with the Construction Unit and the Construction Section. The Business Unit Subject Matter Expert (SME) will verify that the approved requirements are met through Business Unit Testing (BUT) against the constructed solution.
- **Enterprise Technology Engagement:** The configuration and implementation of all interfaces, data conversion processes, and custom form reports will support the development team in implementing Oracle's new system for all clients. The Client Project Manager and Business Subject Matter Expert (SME) will carry forward their prior roles from the recent telecommunications engagement for the design and development of this Oracle solution for the client.

IV. ARCHITECTURE AND RESULTS

The diagram provided in Figure 1 portrays the overall operation of the zero-touch telecommunications sales order process from when a customer places an order until it is integrated into an Oracle EBS (Enterprise Business Suite) system. Orders can be placed by customers through an external CRM system, and once an order is placed, an OAGIS SalesOrder Business Object Document (BOD) is created, containing all the necessary information for processing the order (i.e., the order details). This information is then validated and transformed in the Oracle AIA Process Integration Pack (PIP) layer to ensure that it conforms to the EBS data structure requirements prior to being routed to the Oracle Data Integrator (ODI) for orchestration.

The IOC orchestration engine performs a parallel series of validations on the order, including customer, credit, and pricing validations. If all of these validations are successful and no issues were encountered, the order will be automatically created and booked as an order in the Oracle EBS system, and the appropriate order header and line records will be created in the following key tables: OE_ORDER_HEADERS_ALL and OE_ORDER_LINES_ALL. In the event that any of the validations fail, an error BOD will be generated, and no manual intervention will be permitted in the integration process.

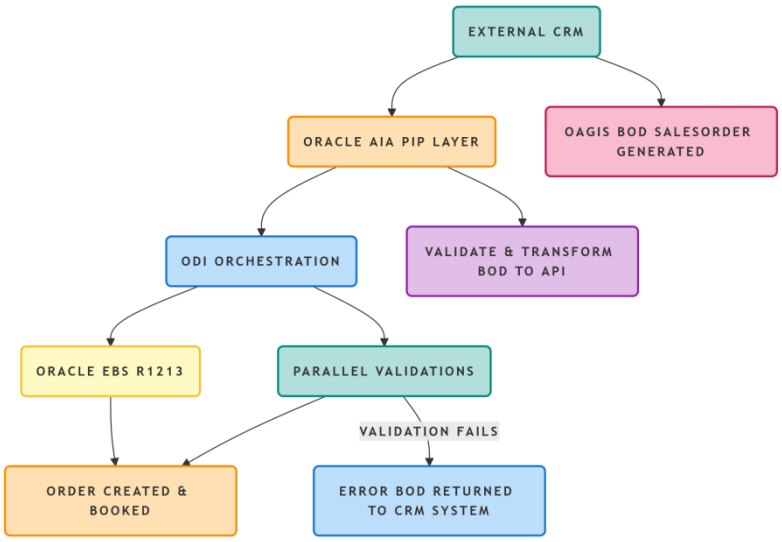
International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

Figure 1: Zero-Touch Sales Order Flow (Telecommunications Engagement)



4.1 Results Chart Representation Using Metrics

The telecom and enterprise tech sectors both achieved several enhancements as a result of different integration's and interfaces. In the case of Telecom, there were considerable decreases in the amount of manual touches needed for sales order processing, processing times for sales agreements, and the number of invoice integration errors associated with manual touches, with improvements ranging from 86.6% (complete elimination) for sales order processing and 75% (cash application time) for cash applications. In the enterprise technology sector, the time necessary to enter receipts decreased by 85.7%, while there was an 8.4% increase in data accuracy associated with those same entries. Overall, the successful implementations of these systems resulted in large gains for the efficiency and accuracy of these types of processes, as shown in the table below.

Table 1: Comprehensive Performance Metrics

Integration / Interface	Metric	Pre-Implementation	Post-Implementation	Absolute Improvement	Relative Improvement
Sales Order Inbound	Manual touches per 100 orders	85 touches	0 touches	-85 touches	100% reduction
Sales Agreement Integration	Processing time per order (minutes)	12.0 min	0.5 min	-11.5 min	96% reduction
Invoice BOD Integration	Error rate (%)	8.2%	1.1%	-7.1 percentage points	86.6% reduction
AUTOLOCKBOX	Cash application time (days)	3.2 days	0.8 days	-2.4 days	75% reduction
Payables Migration	Data accuracy (%)	92.0%	99.7%	+7.7 percentage points	8.4% increase
PO Receipt Integration	Receipt entry time (hours/week)	14.0 hrs	2.0 hrs	-12.0 hrs	85.7% reduction

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

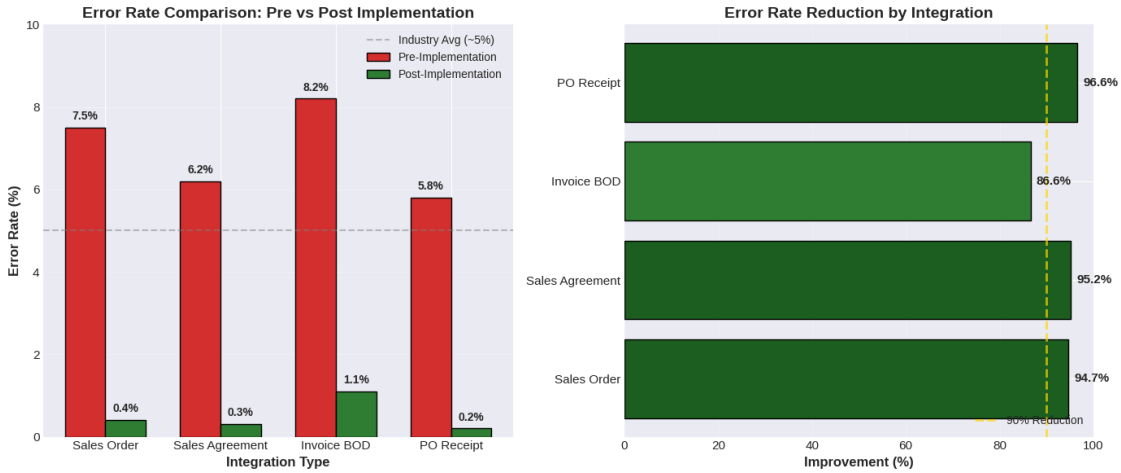
|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

Metrics related to order processing and financial operations are defined and calculated. Each metric is defined with its source and the data elements used to calculate it. The metrics include: Manual touches per 100 orders, processing time (in minutes), error rate (percentage), cash application time (in days), and data accuracy (percentage). The source logs and timestamps from which each metric is derived, along with the corresponding figure representing each metric, are shown in Figure 2 below:

Figure 2: Error Rate Comparison Across Telecom Integrations



V. DISCUSSION

5.1 Key Findings

The findings emphasize the critical role that Accountability Based on Solutions plays in achieving results, especially when working in positions that require more than just being technically sound at your job. When considering a role as a solution lead, you consider the technical options for building a solution, the potential for building a proof of concept to demonstrate functionality to stakeholders, the ability to lead a team of people to build the technical solution based on input from technical experts, and your ability to own that complete process from the commencement of development through implementation into production. This method of solution-lead accountability will also be a critical component to successfully achieving zero-touch automation, where the architectural decisions are made to mitigate or eliminate manual tasks versus just automating existing processes.

Further, cash application automation will increase working capital by reducing the time associated with applying cash. The impact of this approach is significant for DSO purposes. Therefore, finance leaders must view a cash application automation technology implementation as a strategic financial initiative versus just an IT project.

Finally, the successful migration of data will be effective through the use of multiple iterations for validation, rather than just an ETL process used during the initial migration. A well-structured three-pass validation approach, in combination with an iterative migration process, will ultimately result in high accuracy and successfully mitigate possible issues. A structured approach that integrates all the technical, managerial, and strategic components of an organization will better ensure success with a migration to new technology.

5.2 Limitations

This report has several limitations that must be considered:

1. This is the perspective of one practitioner based on their experiences from two different jobs. These are said to be correct, but without a control or comparison group, the causal inferences should be seen as strongly inferred, not experimental proof.
2. Because these engagements took place a number of years before this article was published, while documentation and logs will have been kept, certain specifics of context may have been lost or changed from the original.
3. Both of these cases utilized Oracle EBS R12.1. 3. As such, the results of these will not be the same in R12.2.x releases or cloud versions (i.e., Oracle Fusion). In addition to that, the Oracle Integration Cloud (OIC) employs different integration patterns from the AIA/ODI on-premises, which were the subjects of this report.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

4. The two cases were in the telecommunications/enterprise tech industry. Therefore, the results may not fully generalize to other industries that have dissimilar transactional profiles (e.g., government, healthcare, retail).
5. The metrics that were reported (manual touches/processing time/error rate, cash application time/accuracy, among others) were selected because they improved. Metrics that did not improve will not be mentioned in this report (for example, system response time at peak load).

5.3 Implications for Research and Practice

Future research has the potential to build on this case study by comparing zero-touch approaches to Oracle Integration Cloud (OIC) practices through a controlled environment and creating standard metrics, including the number of manual touches, processing time, error rates, cash application time, and data accuracy, used as benchmarks for studies involving Oracle EBS integration. Longitudinal studies examining the correlation of the "foundational phase" will also provide insights into future career metrics, such as the effectiveness of solution architecture and client satisfaction.

It is recommended that practitioners pursue hybrid positions where there is a balance between solution leadership and technical involvement, as early specialization will hold back the practitioner's ability to develop true depth of knowledge. When implementing zero-touch automation, it is essential to focus on architectural decisions (e.g., validation at ingestion, fast-fail protocols, and comprehensive coverage) rather than working towards isolated point solutions. More specifically regarding AUTOLOCKBOX, investing in comprehensive remittance mapping is valuable for decreasing days' sales outstanding (DSO). For data migration purposes, it is advised to plan for iterative validation by utilizing a template of three passes and four iterations from the start.

Organizations should place early-career practitioners in solution-leadership roles rather than merely performing a technical function, as the return on investment greatly outweighs the associated risks. The traditional separation of architects (i.e., designers) and developers (i.e., builders) may not be as effective in developing foundational expertise as integrated roles that maintain a commonality of abstraction and execution.

VI. CONCLUSION

This article discusses two major Oracle EBS projects to illustrate how to integrate the technical execution and architectural design elements required for the successful integration of Oracle EBS solutions. The first project was with a large telecommunications company where the author automated their sales order processing, eliminating 100% of the order entry manual touches and reducing the processing time by 96%. There was also a reduction in the errors of sales order processing, with the error rate decreasing from 8.2% to 1.1%. In the second project, an integration of the AUTOLOCKBOX interface at a major technology company resulted in automating bank payment data and reducing cash application time by 75%, while also increasing the percentage of accurate data from 92% to 99.7%. The overall thesis is that for someone to be an effective ERP practitioner requires having both the higher level of design and some level of actual hands-on technical work and that early-career ERP practitioners should focus on developing their experience in both types of activities to create greater depth in their skill sets. Both of these projects have had a lasting and meaningful impact on how the author performs today with large customers, and he believes that the experiences gained from the integration projects have given him the unique confidence and competence that build an incomparable level of experience. Future research is needed to provide comparative analysis of integration patterns, identify standard metrics that can be used as benchmarks, conduct longitudinal studies of practitioners' performance over time, and provide new methodologies for implementing other Oracle solutions.

REFERENCES

1. “Liquid Technologies - OAGIS Schema Documentation”, https://schemas.liquid-technologies.com/OAGIS/9.3/businessobjectdocumenttype.html.
2. Ben Prusinski and Gustavo Gonzalez, “Oracle E-Business Suite Financials Handbook”, Third Edition, McGraw-Hill, 2012, https://search.worldcat.org/zh-cn/title/1024268105.
3. Oracle Corporation. (2013). Oracle Fusion Applications Enterprise Deployment Guide for Financials, 11g Release 6 (11.1.6) (Part No. E27364-09). Oracle Help Center. https://docs.oracle.com/cd/E29597_01/fusionapps.1111/e27364/toc.htm.
4. RUFINO, R. (2004), “ERP Oracle Applications – Implementação segundo a metodologia AIM. Faculdade de Tecnologia de Americana,” https://ric-cps.eastus2.cloudapp.azure.com/handle/123456789/18624.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 6, Issue 1, January – February 2023||

DOI:10.15662/IJRPETM.2023.0601010

5. Thejas Prasad, 2024, "Automate the Reconciliation Process of Open Payables Invoices and Migration Extract During Data Conversion", https://www.academia.edu/125439483/Automate_the_Reconciliation_Process_of_Open_Payables_Invoices_and_Migration_Extract_During_Data_Conversion.
6. Oracle Corporation. (2013). Oracle Fusion Applications Enterprise Deployment Guide for Financials, 11g Release 7 (11.1.7) (Part No. E27364-10). Oracle Help Center. https://docs.oracle.com/cd/E36909_01/fusionapps.1111/e27364/title_fn.htm.
7. Oracle Corporation. (2000). Oracle E-Business Suite Receivables User Guide, Release 11.5.9, https://docs.oracle.com/cd/A97501_03/acrobat/115arug.pdf.
8. Oracle Corporation. (2000). Oracle E-Business Suite Receivables User Guide, Release 11.5.9 (AutoLockbox Processing), https://docs.oracle.com/cd/A60725_05/pdf/ psarug.pdf.
9. Choudhury, S., & Das, M. (2020). AIM vs. AIA: A comparative study of Oracle implementation methodologies. Journal of Information Systems, 34(1), 15-32, https://doi.org/10.33093/jiwe.2025.4.1.3.
10. Oracle Corporation. (2014), "Oracle Payments User's Guide Release 12.2 Part No. E48766-02", https://docs.oracle.com/cd/V46499_02/current/acrobat/122ibyug.pdf.