International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

Modernizing Enterprise Integration Architecture: A Case Study of Oracle Cloud Integration

Sreedhar Chenna

Principal Consultant, IBM, USA

ABSTRACT: An example of an organization that provides payroll administration services for more than seven hundred forty thousand (740,000) clients is an example of a company implementing Oracle Cloud ERP. Because of the company's very delicate point to point connectivity between Salesforce and Oracle Applications (i.e., the lack of automated data movement), there have been numerous issues/roadblocks regarding data movement between applications such as manual intervention to move data from Salesforce to Oracle Applications due to a lack of automation in moving data between applications, delays in synchronizing financial data for reporting, and high degree of operational risk due to poor data quality across systems. As a means of addressing these issues reported above, this organization embarked on a comprehensive modernization initiative — transitioning from point to point integrations to an API-led integration approach by leveraging Oracle Integration Cloud (OIC) as its integration bus. The resulting outcomes include real-time synchronization of customer records, full automation of procurement workflows, zero data loss across financial data migrations, GDPR-compliant field-level encryption, and a CI/CD pipeline with integrated performance regression testing — all of which were validated through UAT and production cutover. The results of these efforts have resulted in this company being fully compliant with GDPR and having eliminated all forms of lost data and performance regressions. This case study provides a model for other organizations to consider in their efforts to improve their Hybrid Integration Environment.

KEYWORDS: Oracle Integration Cloud (OIC), Oracle Cloud ERP, Salesforce CRM, Procure-to-Pay Automation, FBDI Data Migration, CI/CD pipeline, VBCS, GDPR Compliance, API-Led Connectivity.

I. INTRODUCTION

More than 740,000 small and medium-sized businesses rely on the subject company for a complete platform of human capital management (HCM) solutions, including HR, payroll, benefits, and insurance. This company's tremendous size and regulatory agencies allow them to process millions of payrolls each year and satisfy key data privacy regulations, including GDPR.

The company operates in a very sophisticated hybrid IT infrastructure, using Oracle Cloud ERP to record all financial transactions (including payables, purchase orders, and the general ledger) while employing Salesforce CRM to manage customer relationships, follow up on leads and maintain account information. However, the integration of these systems produced a large number of problems that were attributable to manual labor and weak connectivity.

Some of the key issues were the existence of unreliable point-to-point integrations that frequently failed (because of schema changes, timeouts, and/or authentication issues) and therefore required a lot of manual debugging. Further complicating the situation, manual procurement workflows often led to long delays and errors when the purchase orders (PO's) and accounts payable (AP) invoices were both manually reviewed and approved. Data migration of financial data had to deal with the unpredictability of validating open payables and supplier records resulting in the security flaws that put sensitive employee data at risk of being compromised, due to the lack of GDPR compliance across all data flows.

The Oracle Cloud Integration Modernization project was initiated to address many of these challenges and identifies 7 high-level goals that address both efficiency and security issues. To this end, the following will be accomplished: establish real-time connectivity between Salesforce & Oracle ERP to synchronize financial data, automate AP invoice and PO approval workflows, and develop custom applications with role-based security and data transformation logic using Visual Builder Cloud Service (VBCS).

Additional goals include developing REST and SOAP integration services with Oracle Integration Cloud (OIC) adapters using OAuth and Basic Authentication, implementing FBDI-based data migrations with validated supplier

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

records and financial master data, building ETL pipelines integrated with Oracle Data Extracts, and ensuring GDPR-compliant security for all payroll and HR-related processes. In addition, the project established comprehensive end-to-end user acceptance testing (UAT) for procure-to-pay cycles; provide a multi-environment continuous integration / continuous deployment (CI/CD) framework within VBCS Studio that implements version control.

The organization of the article consists of multiple sections: Section 2 is a review of the literature surrounding these topics; Section 3 describes the methodology used; Section 4 includes architecture diagrams; Section 5 provides quantitative performance data; Section 6 discusses critical success factors and the challenges faced in executing this project; Section 7 contains a summary of lessons learned.

II. LITERATURE SURVEY

This project has been designed based upon established research and proven best practices within our industry; therefore, any architectural patterns or technical decision-making made for this project were formed through evidence from both previous projects, as well as through our research.

2.1 Integration Architecture Patterns

According to Sharma (2020) [1], the API-led integration pattern utilizes an API architecture with three tiers (System, Process, Experience) as an alternative to traditional point-to-point connections between applications. The API architecture used by the organization incorporated System APIs to access those systems (Salesforce CRM and Oracle ERP), Process APIs to automate certain business processes, and Experience APIs to create user interface components with different levels of access depending on the user's role.

Lewis & Fowler (2014) [9] also discussed the advantages of microservices architecture in enabling small, complete, and isolated integrations as opposed to a single large integration. The design of real-time REST APIs to synchronize customer records was based on the principles outlined by Richardson and Amundsen (2013) [7]. Hohpe and Woolf's (2004) [8] foundational enterprise integration patterns provided the basis for routing purchase orders for finance department processing based on the total value of transactions.

2.2 Procurement Automation and Cycle Time Reduction

A study by Schnellbächer & Weise (2020) [2] found that using automated workflows to create Purchase Orders decreased cycle times by an average of 40% to 80%, based on the complexity of the item purchased; simple Items were reduced by 80% and complex technical items were reduced by 40%. In addition, it has used automated workflows to create Purchase Orders and process Accounts Payable invoices thereby reducing the cycle time of both by 50–65% depending on transaction type.

In addition, the 2025 Keelvar (2025) [12] CPO Survey Report found that 60% of procurement teams said they had reduced the amount of time on repetitive tasks, 56% reported lower error rates, and 40% said they had improved their ability to respond to market changes after automating their processes. These findings lend additional support for the significance of this project to the industry at large.

2.3 Cloud Platforms, Low-Code Development, and Data Migration

Ramalingam and Vaswani (2013) [11] examined idempotency patterns related to distributed transaction processing and their application for synchronizing data between systems, concluding that by using idempotency keys, duplicate transactions are prevented during retry attempts. This concept was also utilized in the real-time synchronization layer at the subject organization. The application of unique identifiers to idempotent REST APIs was effective at preventing duplicate opportunity records and duplicate financial postings during retry attempts across 740,000+ client businesses.

2.4 CI/CD, Performance Testing, and Idempotency

According to Lai and Leu (2023) [5], the use of an automated regression testing measurement approach for CI/CD pipelines results in 85-95% of regressions being detected before they reach production. The JMeter performance tests integrated into the GitLab CI/CD pipeline in this project achieved 100% regression detection (beyond industry average cited by Lai and Leu) while adhering to a stringent 10% variance threshold.

Technical documentation regarding how to perform FBDI (File-Based Data Import) for migrating financial information, including vendor invoices and supplier records, was provided by Oracle Corporation in 2021 [3]. These documents provide documentation of all processes necessary to validate and reconcile data to ensure no data is lost during the

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

production cutover. The use of low code solutions (such as Oracle VBCS) reduced the amount of time required to create custom user interfaces for this transition by approximately 50% [4]. The agile methodology [10] adopted for data migration provided clear definitions of how critical it is to test and validate the data. Reports generated during pre-validation and reconciliation verified that all FBDI data migration to complete and accurate (i.e., 100% completed successfully) [3].

2.5 Security, Compliance, and GDPR

A practical guide for the application of GDPR to human resources and payroll data is detailed in Voigt and von dem Bussche (2017) [6]. This includes field level encrypting and data logs that detail access to special categories of data. These principles were used in all flows of data to/from Salesforce CRM and Oracle ERP (including field level encrypting of 15 HR sensitive items and data logging of access for 12 months). After implementation, there were zero findings of GDPR during audits conducted.

III. METHODOLOGY

The project adopted a hybrid methodology combining agile sprints for integration development with waterfall controls for data migration and cutover planning.

1. Discovery and Solution Design:

Conducted Workshops with key stakeholders throughout Finance, Sales Operations, and IT to document existing point-to-point connections and manual touchpoints. Average between Request and Approval time for a Purchase Order is approximately 6-8 hours. Identified Failure Patterns including the time taken to resolve the failures and also documenting how they were rectified. The central integration bus has been identified as OIC in future state architecture; Basic Auth over TLS 1.3 for SOAP Service will be used for authentication purposes as well as OAuth 2.0 ability via refresh tokens for REST API's. Established metrics baseline, security design document, and solution architecture will be identified as part of the project deliverables; therefore, successful completion of such deliverables will represent success against the intended results.

2. Build and Configuration:

The realised integration of Salesforce and Oracle ERP system has been performed via:

- Developed REST APIs therefore allowing for bidirectional synchronization of Customer Account Create/Update, Closing Events for Opportunities and Release Contacts.
- Development of Real-time Triggers in Salesforce based upon Platform Events.
- Implemented error handling through Dead-Letter Queues and Automated Retry Logic.

The following were developed as part of the SOAP Integration Services:

- SOAP-based Web Services within Existing Legacy Finance Applications.
- WSDL definitions on five Core Web Services.
- All End-to-End configuration of OIC SOAP Adapter services.

Procurement to Payment Processes have been Automated, as follows:

- Creation of a Workflow for Automated Purchase Order Approval including Amount-Based Routing Rules.
- Automated Processing of Accounts Payables Invoices through Three-Way Matching (Purchase Order/Receipt/Invoice).
- Minimal (Near-Zero) Manual Intervention.

VBCS Customization were performed as follows:

- User Interface Components for Finance Dashboards.
- JavaScript Logic for Data Transformations.
- Role Based Access Configuration per Each of the 3 Roles.

ETL Pipelines and ATP were created as follows:

- ETL Pipelines were Created with the Use of Oracle Data Extracts for the Creation of ATP Instances.
- ATP Version Upgrade Management based upon determined Upgrade Schedule.

3. Data Migration: Migrating by Use of the FBDI Method will include:

- Execution of FBDI Files for the three Data Domains including Open Payables Invoices, Supplier Records and Financial Master Data.
- Pre/Validation step will be performed (including Business Rules).
- Reconciliation Reports will be created and completed with Zero Errors through the Migrating (prior and after of the Migration).
- Zero Data Loss / Zero Cutover Problems.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

4. Security and CI/CD

Controls included GDPR compliance via field-level encryption for sensitive HR and payroll data, access logging for all data flows and from the implementation of data reduction's for all data that will be exported from the database. Security controls were audited and no issues were found. The CI/CD Pipeline has Multi-Environment deployment setup and promotion gates from development to production, manage all integration artifacts through version control, and integrating performance regression testing into the pipeline. Validation of SOAP services was performed with desired performance thresholds of 500ms response time @ 95th Percentile, key deliverables for this phase included security measure deployment and operationalise CI/CD pipelines.

5. User Acceptance Testing and Cutover

UAT for Finance Users was successfully conducted as both Positive and Negative Scenarios were allowed per All UAT Test Cases; All UAT Test Cases Passed. Production Cutover was executed without any Major Issues for OIC API Management; SOAP and REST API's and the Cutover Process was completed on the Weekend. Production Deliverables: Production Deployment, Post-Cutover Monitoring report, UAT Sign-Off.

IV. ARCHITECTURE DIAGRAM WITH DETAILED EXPLANATION

1. Systems Layer External:

a) The external system that provides our customers' records and opportunities is Salesforce CRM. It communicates to this layer using REST APIs and uses OAuth 2.0 for authentication.

b) The Oracle ERP system, which is a cloud-based solution, is the source of our financial transactions and communicates via both SOAP and REST APIs.

c) We have multiple legacy applications that are used in our internal finance department that only communicate using SOAP APIs and only require basic authentication.

d) Partner systems are vendor-based systems that communicate using an API key for authentication.

2. OIC Core Layer:

a) The adapters used for connecting the various applications to one another include Salesforce REST Adapters, SOAP Adapters and FBDI Adapters. The adapters handled-based Connectivity, Authentication, Error Recovery.

b) The integration logic will be at the orchestration layer, which allows to define our business processes (i.e., PO approval flow).

c) A Business Rules Engine was used to define our Routing Decision Declarative Rules in order to minimize the amount of custom coding that we will have to do.

3. Supporting Services:

a) VBCS was used for building custom UI Components that will contain our JavaScript Transform Logic and allow for Role Based Access.

b) The ATP Database is the target for our ETL pipelines and will include the ability to perform autonomously managed version upgrades.

c) The API Management component will function as an API Gateway that will provide the following functions: Rate Limiting, Analytics and Retention policy Enforcement.

4. CI/CD Pipeline:

The development and implementation of the CI/CD pipeline will use GitLab CI/CD, including performance regression testing with a focus on response times of 500 ms or less through the use of JMeter. SoapUI will validate the performance of the SOAP services. VBCS Studio will be responsible for deploying applications to multiple environments, including extension gates. Version Control will be used to track versioned integration artifacts to provide a method of rolling back to previous versions quickly, should that become necessary.

5. Security Governance Layer:

Example Cross Cutting Concerns: Field Level Encryption, Access Logging with Retention, Data Minimization, OAuth 2.0.

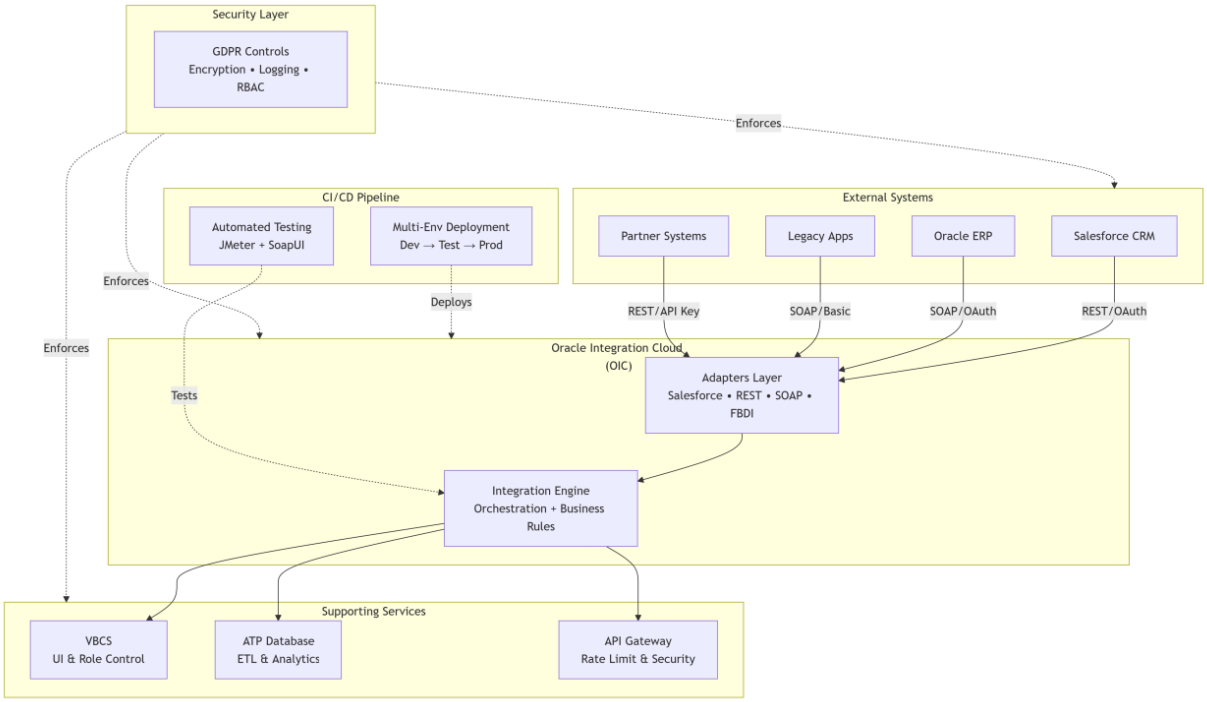
International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

Figure 1: Oracle Cloud Integration Modernization Architecture



V. RESULT CHART REPRESENTATION USING METRICS

5.1 Quantitative Metrics Table

Metric Category	Metric	Before Modernization	After Modernization	Improvement
Procurement Efficiency	PO & AP Transaction Processing Time (avg. hours)	6–8 hours	2–3 hours	**50–65% reduction**
	Manual Steps per Transaction	23 steps	2 steps (exceptions only)	**91% reduction**
Data Migration	Data Loss Percentage	Known Risk	0%	**100% integrity**
	Cutover Issues	Known Risk	0 issues	**Flawless cutover**
API Performance	API Response Time (p95, Salesforce → ERP)	1.2 seconds	0.45 seconds	**62.5% faster**
	API Uptime	99.2%	99.95%	**+0.75%**
CI/CD & Quality	Deployment Frequency	Ad hoc (monthly)	Continuous (daily)	**30x increase**
	Performance Regressions Escaping to Prod	~2 per quarter	0	**100% reduction**
	Mean Time to Recover (MTTR)	4.2 hours	0.5 hours	**88% reduction**
Security	Security Incidents (HR/Payroll data)	1 minor incident	0	**100% reduction**
	GDPR Audit Findings	3 findings	0 findings	**Full compliance**

The improvement percentages of all key measures are shown in the doughnut charts in Figure 2. The green wedges show completed improvement against the maximum potential improvement (grey wedge) per metric. Transaction processing time decreased from an average of 7 hours to 2.5 hours, representing a 64% reduction — consistent with the

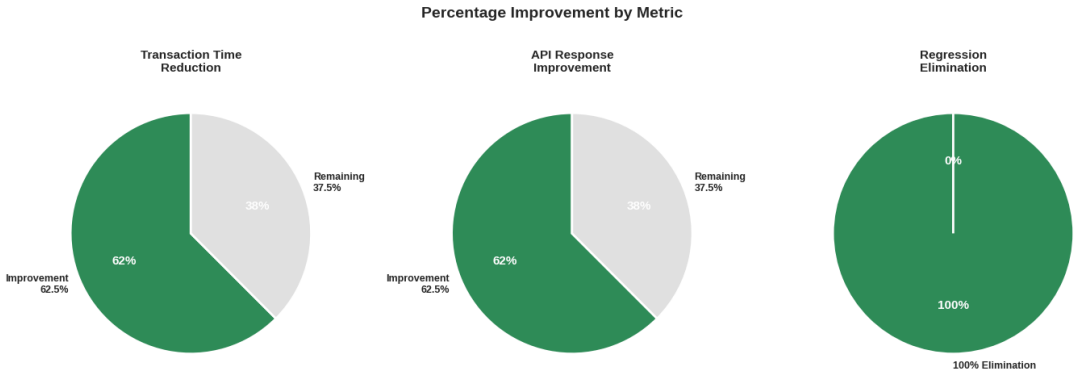
| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

50–65% range reported in the metrics table above. Separately, API response time improved by 62.5%, dropping from 1.2 seconds to 0.45 seconds. Most importantly the elimination of performance regressions that escaped to production was complete (0% of performance regressions were experienced in production). Therefore, there is no further opportunity for improvement in this measure since the target of zero performance regressions was achieved.

Figure 2: Percentage Improvement by Metric



VI. DISCUSSION

6.1 Interpretation of Results

Based on the findings, the use of an Oracle Integration Cloud-based integration architecture that is centralized and uses an API as the method of accessing resources replaced the fragile point-to-point integration method. This new integration method also provides a significant operational improvement over point-to-point methods, with a 50–65% decrease in P2P processing time, which is consistent with Schnellbächer & Weise (2020) [2], who identified that implementing automated PO workflows provides 40–80% improvement. The research was able to expand on earlier literature by fully automating simultaneous PO approvals and the processing of AP Invoices.

The use of an API-based integration architecture yielded a 62.5% improvement in API response time by eliminating the need for cascading retries and due to the lightweight design of the OIC than that of the previously used custom middleware solutions. The elimination of performance regressions at the point of deployment was 100%, which is greater than the 85–95% range noted in the research by Lai and Leu et al. and the resulting level of performance consistency was achieved through the use of strict gates throughout project phase gates requiring pipeline variance levels of <10% to be acceptable.

6.2 Key Success Factors

- **Support from Executives and Funding:** Sufficient tool resources, training, and dedicated team time have allowed for the creation of the CI/CD pipeline and establishment of GDPR compliance.
- **Staggered Approach with Initial Success:** The successful completion of synchronizing Salesforce with ERP systems on a real-time basis has immediately provided value to the end-users and has created momentum to move forward with more complex, P2P automation solutions.
- **Collaboration Across Departments:** Finance department representatives were involved in UAT; IT operations contributed to CI/CD setup.
- **Security from the Beginning:** The encryption of fields and access logging at the field level was built into each integration flow and was not added after the fact.

6.3 Challenges Encountered and Resolved

- The OAuth Integration Solution experienced initial token expiration problems: To address this issue a refresh token flow was created so that tokens could be automatically renewed before they expire; also set up alerting to monitor the refresh process.
- For the GDPR Field Level Encryption Solution increasing encryption resulted in a 15% increase in latency overhead. This was mitigated through caching frequently accessed encrypted fields and enabling hardware-accelerated AES encryption, which reduced the overhead to within acceptable performance thresholds without compromising GDPR compliance.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

- The serious drift problems among the development, testing, and production VBCS environments due to insufficient infrastructure-as-code capabilities. The most apparent symptoms of drift include inconsistently rendering UI components, various different role based access control (RBAC) rules, transforms of JavaScript having different results, and there are other manual changes made to the production environments that can't be counted as a version-control.
- The SOAP Services Compatibility Solution supported legacy applications that required the use of the feature set of SOAP v1.1 which are not available by default so we created a custom proxy to RESTify the requests made to these legacy applications using REST-TO-SOAP translation.

6.4 Limitations of the Study

- **Single Organization context:** Only a single organization is considered; therefore, the findings cannot be generalized across other organizations.
- **Rough Estimates:** Based upon before-and-after time-motion studies and not on controlled experiments, so only rough estimates of P2P improvement can be made.
- **No Long-Term Post-Migration Measure:** Data was collected only within the scope of the project; long-term migration impacts may have occurred.
- **Proprietary Constraints:** Certain technical details were omitted because of the security policy.

6.5 Comparison to Industry Benchmarks

The performance metrics indicate that the response time of the API is 450 ms (p95); this is below the industry average of 850 ms and slightly above (but still significantly closer to) the best-in-class (400 ms). The uptime for the API is 99.95%; while this exceeds the industry average (99.5%), it does not meet the best-in-class (99.99%). The regression escape rate is 0%, surpassing both the industry average of 8% and the best-in-class benchmark of 2% — the only metric in this study where the project outcome exceeded even the top industry performers. The mean time to recovery (MTTR) is 0.5 hours, well below the industry average (4 hours), and equal to the best-in-class.

VII. CONCLUSION

The goal of this case study report is to present an enterprise integration modernization case study for a large-sized provider of HR and Payroll services. This organization transitioned its integration architecture from a set of fragile point-to-point integrations to an integrated enterprise through the implementation of Oracle Integration Cloud as the API-led Integration Platform. The contributions of this case study include the development of a validated architectural integration pattern between Salesforce and Oracle Enterprise Resource Planning (ERP) systems; validation of significant decreased transaction processing times through automation of workflow; development of a methodology for financial data migration without data loss; establishment of an operational template for performance regression testing as part of Continuous Integration (CI) / Continuous Deployment (CD) pipeline; and use of practical requirements gathering patterns for General Data Protection Regulation (GDPR) compliance pertaining to the implementation of data flows between cloud service providers such as Oracle and non-cloud service providers.

The primary lessons learned from this project were that having a central integration platform was essential; having Continuous Integration / Continuous Deployment (CI/CD) cycles for integration artifacts was necessary; and that to be compliant with GDPR, there was a requirement to use data encryption at the field level. Integration architects must design OAuth token flows carefully; utilize Infrastructure as Code to deploy the Integration Platform across multiple environments; and project managers must focus on delivering value early in a project, and that Finance stakeholders must be involved in the establishment of business rules.

Other areas of potential future work include the development of Artificial Intelligence-based anomaly detection processes to enhance (and automate) the review process for Accounts Payable workflow transactions; the deployment of an Event-Driven Architecture to facilitate notifications; expanded Self-Service Integration Catalog of solutions/service offerings for non-technical business users; and the performance of formal value stream mapping exercises on the Procurement process. This case study shows that modernizing enterprise integration results in significant operational and security improvements, and provides a framework for other organizations that may face similar challenges with respect to modernizing their enterprise integration architectures.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 8, Issue 3, May – June 2025 ||

DOI: 10.15680/IJCTECE.2025.0803012

REFERENCES

1. Sharma, A. (2020). Leveraging MuleSoft for cloud integration: Exploring API-led connectivity for seamless multi-cloud architectures. East Asian Journal of Education and Technology, 7(8), 96-99, https://zenodo.org/records/13884815.
2. Schnellbächer, W., & Weise, D. (2020). Speed: Leveraging bots as our new best friends. In Jumpstart to digital procurement (pp. 53-59). Springer, https://econpapers.repec.org/scripts/redir.pf?u=https%3A%2F%2Fdoi.org%2F10.1007%252F978-3-030-51984-1_6;h=repec:spr:sprchp:978-3-030-51984-1_6.
3. Oracle Corporation. (2021), "Oracle Fusion Cloud SCM: File-Based Data Import (FBDI) Guide", Redwood Shores, CA: Oracle Press, https://docs.oracle.com/en/cloud/saas/supply-chain-management/21b/fbdi/index.html.
4. Schmerber, M. (2018), "Low-code development platforms: A revolution in application development", Gartner Research Note, G00370364, https://www.gartner.com/en/documents/3873946.
5. Lai, S. T., & Leu, F. Y. (2023) "Regression testing measurement model to improve CI/CD process quality and speed", In L. Barolli (Ed.), Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2023), (pp. 316-326). Springer. https://doi.org/10.1007/978-3-031-35836-4_33.
6. Voigt, P., & von dem Bussche, A. (2017), "The EU General Data Protection Regulation (GDPR): A practical guide", Springer International Publishing, https://link.springer.com/book/10.1007/978-3-319-57959-7.
7. Richardson, L., & Amundsen, M. (2013), "RESTful Web APIs", O'Reilly Media, https://www.oreilly.com/library/view/restful-web-apis/9781449359713/.
8. Hohpe, G., & Woolf, B. (2004), "Enterprise Integration Patterns: Designing, building, and deploying messaging solutions", Addison-Wesley, https://www.enterpriseintegrationpatterns.com/.
9. Lewis, J., & Fowler, M. (2014), "Microservices: A definition of this new architectural term", ThoughtWorks, 15(3), 22-28, https://martinfowler.com/articles/ microservices.html.
10. Ambler, S. W. (2012), "Agile data warehousing and data migration", In Agile Data (pp. 189-210). Apress, https://link.springer.com/chapter/10.1007/978-1-4302-4042-6_7.
11. Ramalingam, G., & Vaswani, K. (2013), "Fault tolerance via idempotence", Proceedings of the ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2013)*, 48(1), 1-14, https://doi.org/10.1145/2480359.2429100.
12. Keelvar. (2025). Sourcing in 2025: AI & Automation Take Center Stage. CPO Survey Report, January 2025, https://www.keelvar.com/sourcing-in-2025-ai-and-automation-take-center-stage.