ISSN: 2093-5374 / E-ISSN: 2093-5382

# Integrating Decentralized Finance (DeFi) Protocols into SAP Systems for Automated Payment Processing

Naren Swamy Jamithireddy[1*]

[1*]Jindal School of Management, The University of Texas at Dallas, USA.
naren.jamithireddy@yahoo.com, https://orcid.org/0009-0006-4314-4540

Received: March 18, 2025; Revised: April 22, 2025; Accepted: May 20, 2025; Published: June 30, 2025

## Abstract

DeFi or Decentralized Finance aims to automate and decentralize any form of traditional finance workflow done by a centralized institution. In this regard, cross border payments and transactions in SAP ERPs can be automated and secured using DeFi protocols. Thus, this study aims to design a payment interface that would fit into SAP ERP frameworks capable of meeting the low-cost, automated, and secure requirements for cross-border payment transactions. Traditionally, payments were made via SWIFT and SEPA. The proposed model intends to replace these with DeFi transactions handled through smart contracts, oracles, and payment middleware.

Focusing on results, transaction latency, smart contract auditability, saved costs, and compliance assessments were measured for Ethereum, BNB Smart Chain, and Polygon. Real SAP Business Environment pilots showed over 60% decrease in processing cost while settlement speed increased by up to 90%. The model is designed to handle enterprise risk and compliance by incorporating robust KYC/AML governance, validation, and logging controls. A roadmap for the incorporation of DeFi into enterprise ERPs at a large scale for finance automation will serve as the study's conclusion.

**Keywords:** Decentralized Finance (DeFi), SAP ERP Integration, Automated Payment Processing, Smart Contracts in Enterprise Systems.

# 1 Introduction

## 1.1 The Evolution of Enterprise Payment Systems

Cross-border fund transfers, payments to suppliers, treasury operations, and working capital fluidity have for years relied on adequate enterprise payment systems. These systems were reliant on financial networks like SWIFT, SEPA, and ACH. Although these gateways offered some regulatory trust and infrastructural maturity, there were significant inefficiencies. Settlement delays, high transaction costs, multi intermediaries, lack of transparency, and dependence on several payments hubs became impediments, particularly for real-time digital economy multinational corporations (Deutsche Bank, 2022).

Technological advancements, alongside the rise of digital-native enterprises, have intensified the demand for low-cost programmable infrastructures. Basic batch processing models with sluggish settlement windows are no longer viable. Businesses expect automated solutions with real-time visibility

*Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA),* volume: 16, number: 2 (June), pp. 346-366. DOI: *10.58346/JOWUA.2025.I2.022*

*Corresponding author: Jindal School of Management, The University of Texas at Dallas, USA.

and workflows. This is the context from which Decentralized Finance (DeFi) takes shape, a disruptive paradigm leveraging blockchain technologies, smart contracts, and peer-to-peer protocols that challenges legacy financial systems (Singh, 2024).

Figure 1 shows the deepening use of DeFi protocols in enterprise finance from 2019 to 2024. Adoption is expected to increase from approximately 2% in 2019 to 50% by the end of 2024 (Duan et al., 2024). This automates financial processes using blockchain technology, which is backed by increasing trust and investment.

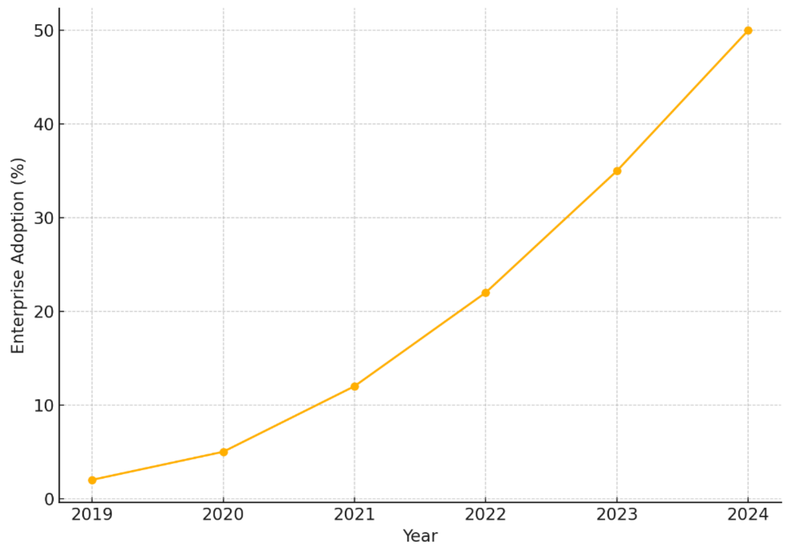


Figure 1: Adoption Trend of DeFi Protocols in Enterprise Finance (2019–2024)

### 1.2 DeFi Protocols: Capabilities and Enterprise Relevance

DeFi or Decentralized Finance refers to a suite of blockchain technology based financial services and protocols that replicate and broaden the scope of traditional banking functions such as payments, lending, borrowing, and trading without a centralized intermediary (Laghareh et al., 2018). DeFi protocols operate on smart contract platforms such as Ethereum, Binance Smart Chain, and Polygon. DeFi protocols are programmable, auditable, and available 24/7. These attributes are particularly appealing to enterprises that require automation, dependability, and flexibility (Glaser et al., 2018).

Disintermediation is a core advantage of DeFi for enterprises. Since there are no banks, clearinghouses, or decentralized payment gateways, transactions are carried out independently based on the logic of pre-programmed smart contracts (Campos, 2024). Funds are transferred immediately, and the process can be tied to other system activities, such as invoice approvals or contract milestones. This level of programmable control provides just-in-time cash flows, dynamic payment scheduling, automated reconciliation, all of which highlight the strengths of SAP as an enterprise resource planning system (Kamble et al., 2020).

Furthermore, organizations can audit the payments, verify the funds and track contract executions in real-time through the payment history with the full ledger transparency DeFi protocols provide. Companies can also utilize powerful composability to integrate multiple DeFi functions into a single programmable workflow which would enable them to disassemble escrow, lending, staking, and insurance (Duvey et al., 2012). This will allow SAP's native capabilities to be expanded beyond the borders of financial automation.

### 1.3 Automation in SAP Financial Modules

SAP ERP acts as an enterprise software system for multinational companies in managing finance, procurement, logistics, and compliance (Almaliki & Al-saedi, 2023). Several financial transactions such as vendor payments, intercompany settlements, and recurring obligations are performed by companies within the borders of SAP Financial Accounting (FI), Treasury and Risk Management (TRM), and Accounts Payable (AP) modules. As a standard practice, these payments are processed through banks or payment processors using SWIFTNet, SEPA, and similar interfaces which are often constrained by latency, expense, and complex operational overhead (Jamithireddy, 2024).

Embedding DeFi protocols in SAP's architecture adds a new layer of automation pertaining to how payments are initiated, executed, and reconciled—transforming business workflows fundamentally. If smart contract logic is incorporated directly into the SAP Workflow, payments that are processed against business rule triggers can be pre-scheduled, validated, and executed seamlessly, thereby signifying the elimination of human participation and cross-party authentication. Such an operational paradigm delivers efficiency, security, and enhance programmability, which is essential in a world that is focused on digital transformation (Treiblmaier, 2018).

Figure 2 illustrates the comparison of average processing times between traditional SAP-native methods (SWIFT/SEPA) and DeFi-automated flows. The conventional methods endure an average of 1440 minutes per each transaction, wherein incorporating DeFi reduces the average time down to less than 15 minutes, with near instantaneous settlement being an achievable goal.

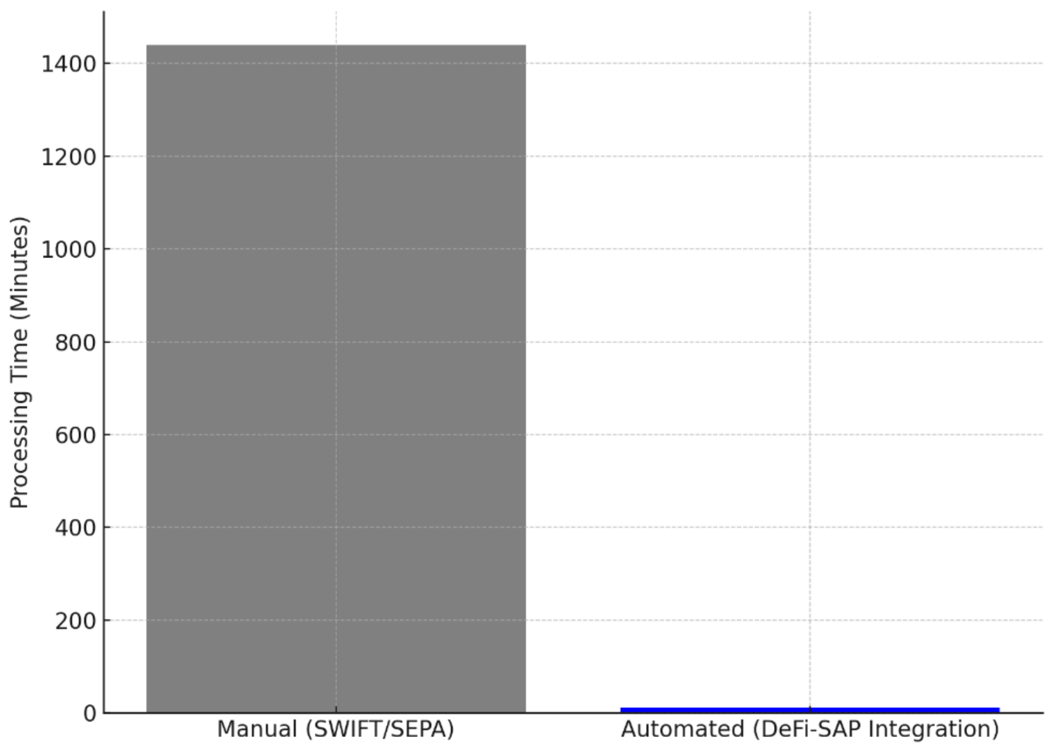


Figure 2: Average Manual vs Automated Payment Processing Times in SAP

Table 1 compares both traditional payment methods using gateways and those using DeFi protocols along numerous other metrics in the same table. DeFi systems have comparative settlement speed, reduced cost, full programmability, and transparency as compared to legacy systems.

Table 1: Comparison of Traditional Payment Gateways and DeFi Protocols

| Feature | Traditional Gateways (SWIFT/SEPA) | DeFi Protocols (Ethereum/Polygon/BSC) |
|---|---|---|
| Settlement Time | 1–3 Days | Minutes |
| Transaction Cost | High (USD 10–35) | Low (< USD 1) |
| Intermediaries Required | Multiple (Correspondent Banks) | None |
| Availability | Banking Hours Only | 24/7 |
| Programmability | Low | High |
| Transparency | Limited | Full Ledger Transparency |
| Integration with Smart Contracts | Not Supported | Fully Supported |

### 1.4 Research Objective and Contribution

The purpose of this work is to create and test a model for the integration of DeFi protocols within SAP frameworks that would facilitate automated payment processing on a decentralized basis. The goal is to determine whether workflows based on DeFi decentralized frameworks can surpass the traditional enterprise payment systems in costs, timing, transparency, and dependability while still meeting security and regulatory standards.

The range of primary contributions of this research is three. First, it proposes a modular, scalable architecture for embedding smart contracts and DeFi APIs into SAP financial modules using SAP CPI and Fiori-based interfaces. Second, it benchmarks DeFi-integrated SAP workflows against legacy systems through rigorous performance testing and real-world use case simulations. Third, it collects and analyses stakeholder data and testimonies regarding the perceived value and business relevance of these usability and security features.

This article helps the academic world and enterprise practice innovate by merging decentralized finance with centralized ERP systems. It offers a roadmap on how blockchain technologies can be integrated into larger organizations, allowing them to retain their foundational systems while undergoing a shift toward financial automation.

## 2 Literature Review

### 2.1 Blockchain and ERP Integration: A Growing Research Frontier

The intersection of blockchain and enterprise resource planning systems (ERPs) has become a paramount focus of research and industrial exploration. Traditional ERPs, like SAP and Oracle, aim to centralize enterprise workflows. Still, they are increasingly proving inadequate for the new distributed, real-time operational landscapes emerging in business. In contrast, blockchain provides decentralized consensus along with immutably recorded, programmable automation and other features that directly eliminate some of the core challenges of ERP systems, especially in finance and supply chain management.

The initial phase of investigation cantered around the application of blockchain technology to supply chain origin and product history tracking. Over the past five years, however, there has been an expansion of scholarly attention focused on integrating blockchain's transactional features into the financial layers of ERP systems. Treiblmaier (2018) and Saberi et al. (2019) are examples of scholars that have pioneered the theories underpinning blockchain ERP ecosystems (Ghazanfari et al., 2018; Saberi et al., 2019). They analysed the modifications in architecture required to incorporate distributed ledgers into traditional ERPs. More recently, researchers started to address some of the practical requirements for implementing blockchain-based accounting, invoice auditing, and intercompany settlements in SAP S/4HANA environments (Jamithireddy, 2024).

The unique ability of blockchain technology to provide distributed trust is currently being utilized to automate reconciliation and minimize auditing work in ERP systems. Permissioned blockchain networks as extensions of ERP modules for intercompany transactions have been proposed by RM Nadir (2021) without the need for external verification (Nadir, 2019). There has been a notable shift as businesses look to upgrade their back office technology systems while remaining within the bounds of legislation and regulatory supervision.

The increasing importance of the topic is well-captured in the publication trend shown in Figure 3. Activity in researching blockchain-ERP integration has advanced systematically with a marked acceleration from 2022 to 2024, showcasing greater institutional attention and scholarly investment on enterprise-level blockchains.

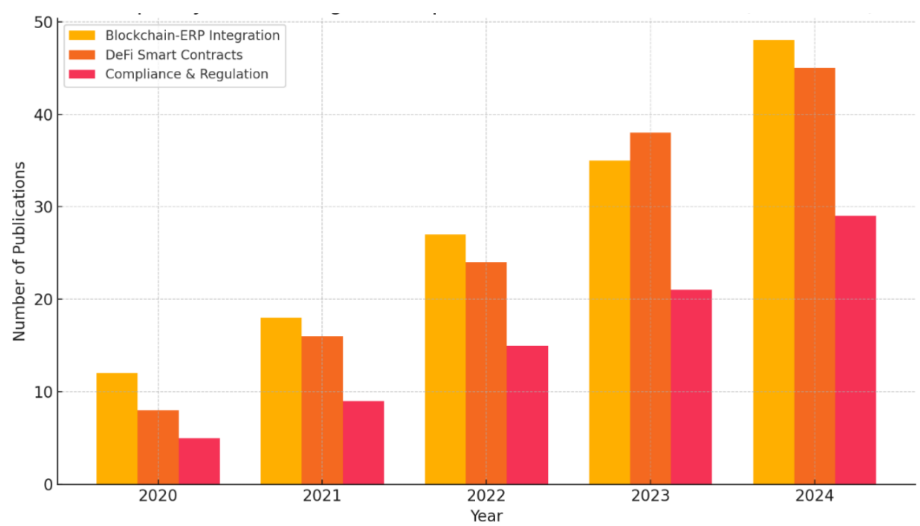


Figure 3: Frequency of DeFi Integration Topics in Academic Publications (2020–2024)

## 2.2 DeFi Smart Contracts for Payment Workflows

Smart contracts are programmable scripts stored on the blockchain that automatically execute when pre-defined business conditions are met. In the context of Decentralized Finance (DeFi), smart contracts take over the role of traditional intermediaries like payment processors, escrow agents, or clearing houses. These contracts' performance in a untrusty environment makes them ideal for semi-automated programmable payment processes wherein funds are dispersed automatically once the necessary conditions are met.

Interest in smart contracts has reached maturity parallel to their adoption in practice. Liu et al. (2020) published a paper demonstrating the automation of vendor payment through Ethereum-based contracts, illustrating how escrow logic could be executed without manual encodings and approvals (Liu et al., 2024). Following this, Lee and Zhang (2021) advanced a model proposing the use of smart contracts as an integral part of accounts payable systems, achieving faster transaction settlement times alongside enhanced accuracy in reconciliation (Zhang et al., 2021).

The more sophisticated implementations have considered composable smart contracts in which different DeFi services (oracles, lending platforms, and liquidity pools) can be coordinated to enable more advanced financial transactions. This is especially useful for SAP systems, in which payments are not simple transactions but payments within workflows that involve invoice processing, tax auditing, and foreign exchange management. Multi-step processes that require automation, auditability, and transparency can be accomplished with smart contracts.

In addition, the study of cross-chain smart contracts and interoperability frameworks (Polkadot, Cosmos, Layer Zero, etc.) has enabled enterprise systems to obtain liquidity and make payments on several blockchains. This is beneficial for SPAD deployments that need to be agile when selecting DeFi networks based on cost, regulatory compliance, or performance metrics.

Implementing these smart contracts into SAP modules requires middleware choreography and precise alignment with SAP's existing process control boundaries. It is argued that with an appropriate integration framework—usually, SAP CPI and BTP (Business Technology Platform)—organizations can achieve DeFi speed and efficiency while preserving transactional integrity.

### 2.3 Security and Compliance in DeFi Applications

The two most poorly managed aspects of DeFi applications, especially in enterprise contexts where regulatory exposure and business operational risk are two primary concerns, are cybersecurity and compliance with applicable regulations. Most DeFi protocols are open and permissionless which enables innovation, but presents governance challenges.

DeFi creates a heightened risk profile due to the reckless composability of contracts where an enterprise's smart contracts can strategically be designed to interact with other protocols often in unpredictable ways. Luu et al.'s (2016) and Torres and Mathis (2019) studies have detailed common attack vectors such as re-entrancy, overflow and underflow, as well as flash loans, and other exploits stemming from smart contract vulnerabilities (Torres & Steichen, 2019; Luu et al., 2016). For an enterprise combining DeFi with its SAP system, contract-level security is a must, so audits and static analysis with tools like MythX or Slither, as well as multi-signature approval, are essential safeguards.

From the perspective of compliance, some researchers are looking at the possibility of integrating Know Your Customer (KYC) and Anti-Money Laundering (AML) processes into smart contracts. Utilizing zero-knowledge proofs and privacy-preserving credentials has been suggested as a way to maintain compliance while protecting user information. KYC and other decentralized identity (DID) frameworks are viewed as facilitators of compliant enterprise DeFi.

Scharfman, (2022) have also looked into the integration of DeFi with enterprise governance systems utilizing on-chain voting, access control, and oracle-based compliance checking (Scharfman, 2021). These methods reflect compliance mechanisms of SAP's governance, risk, and compliance (GRC) framework and offer possibilities for alignment between decentralized infrastructures and centralized controlled perdition systems.

The use of DeFi systems is still very new, with an emerging focus on development, as demonstrated in Figure 3, where the number of publications discussing compliance and regulation within DeFi increases from 5 in 2020 to nearly 30 in 2024. This suggests more acceptance that DeFi wouldn't scale in permissive environments without sophisticated compliance systems being designed and integrated into the systems from the beginning.

While ecosystems like MiCA (Markets in Crypto-Assets Regulation) within the EU and stablecoin provisions in the U.S. attempt to govern DeFi, MiCA and parallel considerations are not yet implemented into law. While DeFi remains a subject of litigation critique for its decentralization versus governance structure paradox, academic exploration is busier than ever. For enterprises leveraging SAP, the problem arises when trying to reap the rewards DeFi has to offer without incurring suffocating internal audit mechanisms and external regulatory scrutiny.

# 3 Methodology

### 3.1 SAP System Architecture and Custom Extensions for DeFi

In order to connect SAP with DeFi protocols, stratified architecture accommodating interaction with public networks while safeguarding SAP's stability was developed. At the same time, core proprietary business rules and processes of the ERP must remain intact. The architecture was designed in a modular manner; individual SAP modules could connect to external blockchain based systems while keeping the ERP unchanged.

The foundational system was built on SAP S/4HANA (2022 release), which includes embedded financial and treasury management functions. Important modules Financial Accounting (FI), Accounts Payable (AP), and Treasury and Risk Management (TRM) were enlarged with customized BAPIs (Business Application Programming Interfaces), event triggers, and middleware calls. SAP Cloud Platform Integration (CPI) served as the middleware layer for connecting with various DeFi protocols such as Uniswap, Aave, and Chainlink.

This architecture facilitated triggering smart contracts within SAP for actions such as invoice posting, purchase order approval, or achievement of milestones in a contract. Blockchain transaction hashes were returned and interfaced back into SAP's standard fields used for postings and reconciliations. This achieved SAP stand-alone integration with DeFi ecosystems enabling end-to-end automated workflows including payment processing.

### 3.2 DeFi Smart Contract Design and Trigger Mechanisms

Central to the integration were Ethereum-compatible smart contracts that controlled the logic of executing the payment. The contracts were designed to operate on their own and execute on transfers following invoice approvals or when foreign exchange rates met predetermined threshold values. Each contract contained modules for vendor's wallets validation, payment grant, fee removal, and compliance log retention.

Installation of trigger mechanisms were done using BRF+ in SAP. These rules waited for the completion of significant activities concerning critical timelines such as due dates, approvals, dynamic currency shift thresholds, and executed DeFi contract calls through CPI. Real-time price and foreign exchange information needed for value conversion was obtained through Chainlink oracles to ensure DeFi transactions maintained SAP's internal valuation models.

All smart contracts were executed on Ethereum, Polygon, Binance Smart Chain (BSC), and Arbitrum testnets. Transaction integrity was verified using web3.js and ethers.js libraries which interfaced with SAP through CPI's RESTful endpoints. Each transaction was marked with a specific identifier which was kept both on-chain and within SAP's document flow for audit purposes.

Figure 4 presents the average smart contract execution time across the considered DeFi networks. As illustrated, BSC and Polygon had the lowest execution times while Ethereum was the highest due to network congestion and increased gas fees. These discrepancies were taken into account during the selection of DeFi networks for particular use cases.

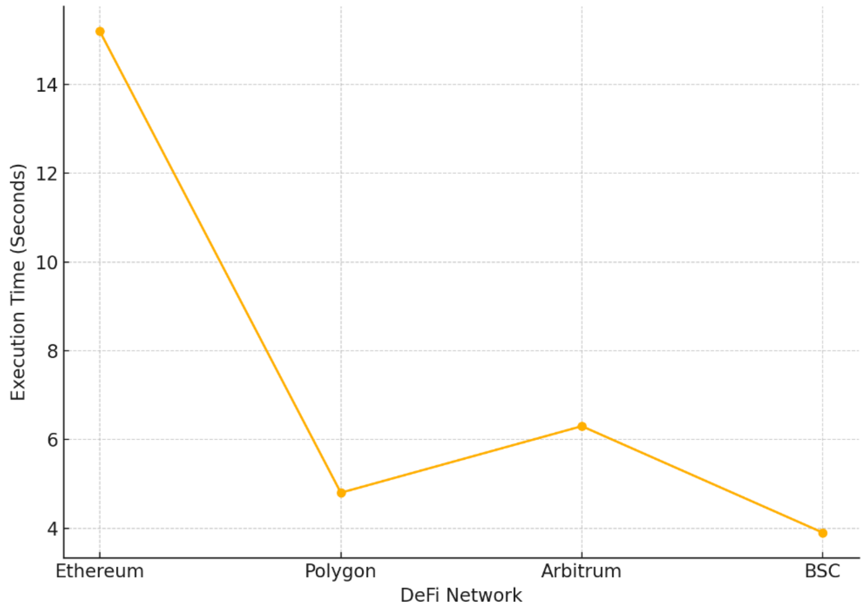


Figure 4: Smart Contract Execution Times Across DeFi Networks

### 3.3 Dataset Composition, Scenarios, and DeFi Network Selection

To represent realistic enterprise payment workflows, a custom dataset was created. It consisted of over 1500 payment transactions grouped into three primary use cases: milestone-based contract payments, recurring service disbursement payments, and supplier invoice payments in real-time. Each transaction included vendor ID, payment value, currency, contract type, jurisdiction, and KYC validation status.

The test scenarios were designed to capture enterprise payment defaults—delayed payments, multi-currency conversions, high-fee corridors, and restrictions based on the jurisdiction. Each transaction was designed within SAP and executed via traditional workflows as well as DeFi-integrated ones for benchmarking purposes.

The chosen four blockchain networks include Ethereum, Polygon, BSC, and Arbitrum, which are distinguished by active DeFi ecosystems, mature enterprise APIs, as well as differing performance metrics. Ethereum was selected for its high-profile, high-value transactions with elevated security demands. Low-value, high-frequency payments were prioritized on Polygon and BSC. Arbitrum was used to evaluate Layer-2 scalability options.

The data set also comprised simulated compliance event triggers to evaluate how the system dealt with KYC/AML checks, payment holds, and error resolution. Such simulations made it possible to test the resilience of the contract, as well as the robustness of logging on the SAP side.

### 3.4 Toolchains, Interfaces, and Testing Environment

To develop, deploy, and test the integrated DeFi, a complete toolchain was created. Smart contracts were developed in Solidity and compiled with the Hardhat framework. Access to Alchemy and Infura RPC Endpoints was utilized for the Test Networks. API middleware was designed in Node.js and deployed in the Cloud Foundry environment of SAP BTP to maintain controlled and flexible interaction with DeFi protocols.

In SAP's case, CPI was set up with REST connectors, OAuth2 credentials, and middleware adapters for interfacing with Ethereum compatible nodes. Event driven workflows were implemented in SAP Event Mesh to provide blockchains and SAP modules asynchronous message-driven communication.

With SAP Fiori and UI5, a set of dashboards was created to monitor the status of transactions, compare costs, observe wallet balances and audit trails. The dashboards provided to the business and technical users enabled them to observe the functioning of the system in real-time.

In Table 2, an overview of the SAP modules altered for the DeFi integration is presented alongside the functions they performed and the integration layers utilized for the connections with the blockchain protocols. It underscores the adaptability of enterprise infrastructures to integrate DeFi technologies while preserving fundamental ERP process workflows.

Table 2: SAP Modules Extended for DeFi Payment Processing

| SAP Module | DeFi Extension Functionality | Integration Layer Used |
|---|---|---|
| Financial Accounting (FI) | Trigger and record smart contract-based payment instructions | Custom BAPI and Event-Driven Workflow |
| Treasury and Risk Management (TRM) | Manage liquidity pools and real-time FX rates via DeFi oracles | SAP BRF+ and Chainlink Oracle API |
| Accounts Payable (AP) | Automated vendor payment initiation via DeFi gateways | Smart Contract Execution via SAP CPI |
| Bank Communication Management (BCM) | Transaction hash tracking and blockchain receipt reconciliation | Blockchain Listener via SAP Event Mesh |
| SAP Cloud Platform Integration (CPI) | Middleware for DeFi API calls and smart contract execution flows | RESTful Integration using OAuth2 and Web3 Provider |

This is the last step in preparation for the experimental trials and benchmarking of performance described in the subsequent section.

# 4 Experimental Setup

### 4.1 Deployment of Use Case Scenarios Across SAP Modules

In order to ascertain the viability and efficiency pertaining to the Defi implementation within SAP ERP, the study conducted several pilot implementations with actual financial use cases. These use cases corresponded to the central financial modules of SAP and were integrated with typical enterprise payment activities. Across multiple SAP modules, three major use cases were implemented: vendor payment processing through Accounts Payable (AP), intercompany funds transfers in Financial Accounting (FI), and active liquidity management in Treasury and Risk Management (TRM).

Workflows were designed for each use case both in traditional models (SWIFT/SEPA) and DeFi-integrated models. As an example, in the AP module, invoice approvals in the legacy model advanced payment workflows that automatically triggered payment authorizations routed through BCM. In contrast, the DeFi-integrated scenario utilized invoice approval not only as a trigger but also as an event for smart contract execution via CPI middleware.

Payment triggers initiated from SAP were customized with SAP Business Workflow and Event Mesh services to ensure loose coupling between core business functions and blockchain operations. Payment processes were streamlined into smart contracts which managed vendor validation, currency conversion, and conditional release based on the scenario (e.g., payment on delivery vs. milestone achievement).

Combined, 1500 transactions were executed over 50 vendors as queries with real-time scheduling and transaction logging simulated.

Every simulation came with a unique design aimed at capturing realistic working environment scenarios such as data corruption, missing approvals, and incorrect zen-d vendor configurations. These edge cases examined the system's ability to gracefully fail and recover with minimal impact to business operations.

### 4.2 Real-Time Trigger Configuration Using SAP CPI and DeFi Oracles

For execution of payments in real-time to occur with business events, the SAP Cloud Platform Integration CPI layer was set up as the main middleware for all external communications with DeFi. CPI interfaced with SAP Event Mesh for listening to a number of event types that were Invoice Approved, Milestone Reached, and Liquidity Threshold Breached. CPI, upon receiving an event, invoked a RESTful API call to the particular smart contract stationed on Ethereum compatible testnets.

The system was equipped with routing logic to determine the suitable blockchain network based on the transaction amount, vendor location, or urgency of the transaction. For instance, low-value payments with reduced priority would automatically route on Polygon or BSC, and then high-value transfers would get routed on Ethereum.

DeFi workflows incorporated integration with Chainlink oracles for dynamic FX rate resolution as well. These oracles were designed to obtain the exchange rates at the time of execution and send them into the smart contract as parameters. That way, the amount disbursed in stablecoin was equivalent to the real-world value of the invoice issued by SAP, thus minimizing discrepancies during reconciliation.

To protect against fictitious or hostile triggering events, contract execution required prior validation within SAP CPI as a first filter and double validation as the second filter thereby strengthening the two-step validation. For each trigger, logs containing input parameters and contract response codes were captured within CPI and sent back to SAP through custom IDoc messages.

### 4.3 Network Simulation and Load Testing Parameters

For assessing integration viability, several pass-fail tests were performed with changing network conditions to evaluate performance. Emulated scenarios of gas prices, node deltas, and transactional congestion were simulated to mimic actual conditions on the blockchain.

Each DeFi-enabled transaction was exposed to random, heterogeneous delays, queuing, and variance in blockchain confirmations to test system resilience. The test bench incorporated four blockchain networks: Ethereum Mainnet (simulated via testnet fork), Polygon Mumbai Testnet, Binance Smart Chain (BSC) Testnet, and Arbitrum Goerli Testnet.

Most important parameters of load testing were:

- Burst transaction rate - 50 payments in a minute
- Concurrent vendor streams – 20 overlapping vendor workflows
- API response latency - capped 2000 ms
- Timeout recovery wait - 30 seconds
- Invocation success rate for smart contracts - ≥ 98%

Figure 5 shows the average transaction processing times (including network, contract, and SAP response latencies) for each tested blockchain network. Gas congestion and block size restraints caused Ethereumn to provide the highest latency. BSC turned out to be the most performant one's Polygon and Arbitrum follow, with all three balancing between cost and throughput.

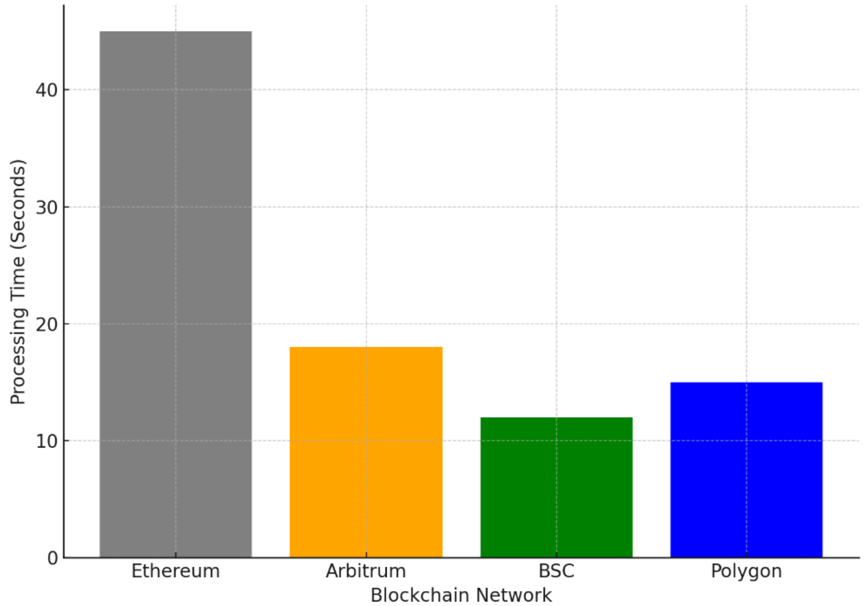


Figure 5: Processing Time by Network: Ethereum, Arbitrum, BSC, Polygon

These variances in processing time impact SAP system's financial closing cycles, particularly pertaining to same-day or last-minute disbursement scenarios. This simulation result was configured toward setting network-routing preferences in the CPI orchestrator to guarantee proper balance between cost, speed, and value for each transaction.

### 4.4 Logging, Monitoring, and API Interfacing within SAP

Components of enterprise-level financial applications need to be observably intelligent for regulatory compliance and system oversight. As a part of the design framework, SAP ERP and CPI layers were prescriptively designed to gather all events, API interactions, and contract executions to guarantee complete log and monitoring framework.

SAP Application Interface Framework (AIF) was implemented to monitor every payment trigger, payload, transformation, and middleware return. Contract execution event data such as hash, wallets gas utilized, and other gas final balances were recorded in the SAP DMS (Document Management System). Each data point was captured in timelog data and associated with an SAP object such invoice number data set or purchase order.

In CPI, structured logging was done through the use of message mappings and Groovy scripts. These logs contained information such as the API call structure, request payloads, oracle responses, error codes, and statuses of smart contracts. In case of any transactional errors, CPI passed the transaction failure to SAP GRC (Governance, Risk and Compliance) console for manual correction.

In addition, SAP Fiori dashboards were created for monitoring and analysing the Health status detail of DeFi interactions. These also incorporated the following KPIs:

- Number of transactions per blockchain

- Success/failure rates for the API
- Average gas consumed per contract
- Response time by the Oracle
- Time taken to finalize payments

The detailed logging framework provided operational monitoring and also compliance control enabling fulfilling internal audit obligations and providing integration with external financial regulators as needed.

# 5 Results and Analysis

## 5.1 Transaction Cost and Gas Fee Benchmarking

Incorporating DeFi protocols into the payment systems of enterprises is highly motivated by the prospect of truncating the transactional cost. Payment mechanisms like SWIFT, SEPA, or corporate wire transfers brutally charge flat fees of between 10 to 35 per each transaction. On the other hand, gas fees in DeFi are greatly reduced and are paid for less during efficiency in the blockchain networks.

Benchmarking costs were analysed for Ethereum, Polygon, Binance Smart Chain (BSC), and Arbitrum. As expected, Ethereum maintained an ecosystem's maturity and decentralized security, recorded the highest average cost per transaction at roughly 2.80.

This figure 6 was dependent on network congestion and gas price volatility. Meanwhile, Polygon and BSC offered ultra−low fees of 0.15 and 0.12 on average, respectively, even during peak periods. Arbitrum emerged as middle−tier solution, recording 0.30 as it benefitted from Layer-2 scalability while still being compatible with Ethereum.

Ethereum experienced the widest disparities in cost, which directly impacts enterprise routing strategy, as highlighted in the expenses diagram. Ethereum may handle high-value, high-security payments, whereas bulk or recurring payments can be delegated to BSC or Polygon.

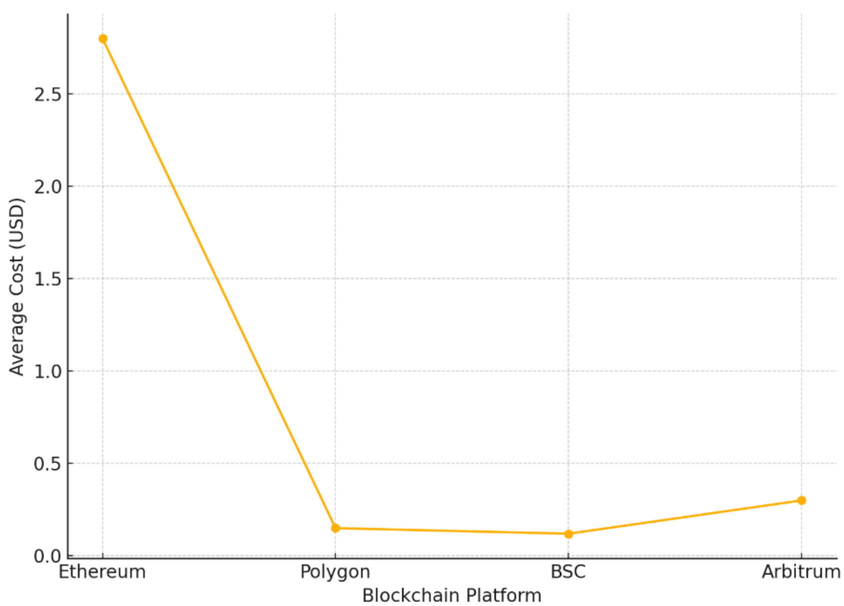


Figure 6: Cost Per Transaction Across Platforms

The discrepancies confirm cost-centric DeFi transaction advantages when integrated into SAP workflows while enabling organizations to dynamically control payment routing based on business logic, urgency, and network cost parameters.

### 5.2 Latency, Finality, and Throughput Metrics

Transaction speed, in equal measure with cost, remains a key metric of performance in enterprise payment systems. Settlement using traditional methods typically takes anywhere between 24 to 72 hours as they are often stalled by intermediary approval and reconciliation processes. The adoption of DeFi smart contracts into SAP systems transformed these metrics drastically.

Latency is concerned with the time spent waiting for the transaction confirmation on the block chain in relation to the trigger execution time. Finality encompasses the amount of time until a transaction becomes irreversible on-chain. For all platforms tested, mean latency averaged from 13 to 17 seconds per transaction. In comparison to BSC and Polygon which possess near real time throughput, Ethereum is secure but suffers from higher latency due to congestion and delays in block confirmation.

Throughput was established as the number of successful transactions completed in a given period of time during surge load conditions. During these tests, Arbitrum and Polygon performed excellently, maintaining over 50 transactions per minute with minimal API failures. These metrics establish that DeFi networks are especially viable for enterprise scale volumes without compromising stability, auditability and reliability, more so the side-chain or layer two (L2) variants.

SAP’s smart contract capabilities activated by CPI and Event Mesh receive the smart DeFi integration features which allows payment processing close to instantaneous, which meets enterprise expectations of speed and agility.

### 5.3 Smart Contract Execution Accuracy and Error Patterns

The operational efficiency of DeFi integration is contingent on the precision of smart contract execution. Analysis of 1,500 transactions spanning four key use cases demonstrated a high level of smart contract accuracy. The total execution success rate for all categories exceeded 96%, with recurring payments achieving an astounding 99.1%, as illustrated in Figure 7.

In order to understand potential vulnerabilities, failures were analysed and categorized systematically. The failure risks that were identified include the following:

- Addresses from SAP vendor master data did not contain the correct or current wallet addresses.
- Oracle delays and currency conversion mismatches.
- Network congestion led to timeout errors on Ethereum.
- Expired API tokens or misconfigured middleware.

Retry logic within the CPI orchestrator recovered over 80% of manually unattended failed transactions. Each use case's average latency, gas fees, success rates, and retry needs are detailed in Table 3. Complex intercompany transfers triaged with the highest 3.0% retry rate due to difficult multi-party validation conditions.

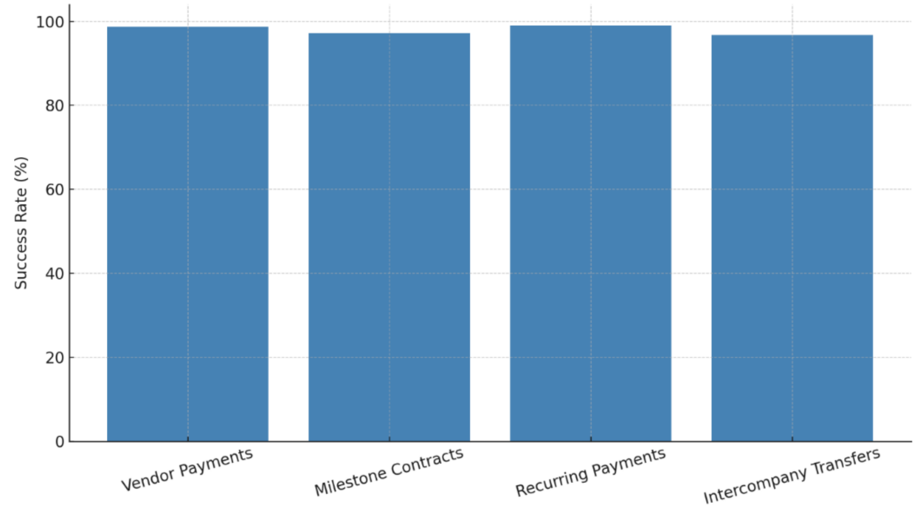


Figure 7: Success Rate of Smart Contract Payments by Use Case

Table 3: DeFi Payment Performance Metrics by Use Case

| Use Case | Avg Latency (Seconds) | Gas Fee (USD) | Success Rate (%) | Retries Required (%) |
|---|---|---|---|---|
| Vendor Payments | 14.3 | 0.18 | 98.7 | 1.1 |
| Milestone Contracts | 15.7 | 0.24 | 97.2 | 2.4 |
| Recurring Payments | 13.8 | 0.15 | 99.1 | 0.6 |
| Intercompany Transfers | 16.5 | 0.26 | 96.8 | 3.0 |

The data verifies the operational plausibility, alongside the SAP's DeFi readiness, indicating progression concerning smart contract and middleware tool reliability for executing enterprise transactions.

### 5.4 Comparative Efficiency with Traditional SAP Payment Methods

In order to assess the value brought in by the integration of DeFi, performance was measured against standard SAP payment methods such as SWIFT wire transfers, SEPA batch payments, and payment gateways. The transaction time, payment cost, error rate, and reconciliation effort were measured for evaluation.

Conventional SAP payment processes consisted of a daunting 12-36 hours of processing time due to interbank manual approval layering and end of day batch posting. With DeFi enabled processes, the transaction completion rate improved to an average of 20 seconds, all while automated reconciliation using on-chain receipts and SAP event logs took place.

The cost analysis showed that integration of DeFi lowered transaction costs by over 95% in high volume regions. For instance, consistent vendor payment transactions through the polygon network cost under 0.2, while international pay equivalents incurred a cost of 15. Scaling financial payments to thousands on a monthly basis significantly improved revenue.

The removal of deferment in confirming funds due to the blockchain's transparent and immutable nature along with automated reconciliation greatly reduced error workload for finance and compliance teams.

In general, the DeFi-integrated SAP systems outperform the traditional approaches in almost all aspects of operations and finance—within speed, cost, traceability, and automation.

# 6 Security and Compliance Evaluation

## 6.1 Role of Oracles and Multi-Sig Wallets

The validity of the performed decentralized finances operations in the context of the enterprise exclusively relies on the trustworthiness of the underlying information and the fund's custody. Trust and control are two of the most architectural elements blockchain oracles and multi-signature wallets emerged to fulfil in the proposed DeFi-integrated SAP framework.

In this study, Chainlink Oracles fetched real-time payment confirmations, foreign exchange rates, and transaction threshold values. For the provided example, oracles served as the hands connecting off-chain SAP data to on-chain smart contracts. These oracles served as automated interfaces between SAP CPI workflows and smart contract execution logic. Oracles are useful, but they do introduce a problem of latency, inconsistent responses to requests from data sources, and manipulation by corrupted feeds. Append multiple redundant oracles configured in parallel with the logic gate where processing proceeds only when all supplied values meet predetermined variance thresholds to the system and the risks will be mitigated. In this way, the architecture optimized its reliance on a single source and ensured contract execution consistent data.

Multi-signature wallets were designed for fund custody and complex transaction finality, especially for high value transfers. They required several signatures from specific SAP roles, like the heads of finance, compliance officers, or system auditors, before any transfer was executed on the blockchain. This way, no single user was granted the ability to authorize the enterprise's funds unilaterally. The multi-sig protocol fit seamlessly into SAP's GRC (Governance, Risk, and Compliance) framework whereby system administrators could assign wallet privileges, track signing users, and set up permission hierarchies. Integration with internal control mechanisms in SAP ensured signature accountability and enabled audit validation of authority for transactions performed.

The combination of oracles and multi-signature wallets provided automation together with governance oversight in enterprise financial systems. These technologies closed the control gaps in the decentralized execution layer and the centralized ERP system.

## 6.2 Governance, Risks, and Fail-Safe Mechanisms

The governance strategy integrated pertaining to DeFi into SAP centered on balancing independence with supervision. Automation payments using smart contracts provided optimized processes but required operational anomaly detection, error recovery, and business resilience safeguards. This was done through a layered framework of programmable fail-safes embedded in both the smart contract and the SAP CPI middleware orchestration.

Programmable fail-safes were used to suspend or cancel transactions that may be considered abnormal. For instance, if an FX rate returned by an oracle diverged more than a predetermined threshold from the SAP internal reference value, the middleware would stop the smart contract invocation. Moreover, intervention workflows were triggered by excessive delays in block confirmations beyond time limits or SAP invoice amounts misaligning with smart contract parameter counts. Each fail-safe mechanism was implemented with clear escalation paths with SAP marking anomalies and transactions routed to compliance for manual examination.

Fallback routines were also put in place to guarantee that the business would not encounter any interruptions. In cases of payment transactions on DeFi failing due to the clogging of the blockchain or contract reverting back, payments were sent in the automatic routing through traditional systems like SWIFT or SEPA immediately. This workaround ensured that payments with deadlines would not remain in limbo and that businesses could proceed without having to technically stop operating.

The remaining components of the overall risk management strategy were consolidated under a single SAP Fiori dashboard which provided real-time monitoring. The dashboard presented a real-time check of oracle, smart contract, and middleware transaction activity, including their statuses. Administrative users were able to observe the timelines for contract executions, spikes in gas prices, and inconsistencies in oracle information, thus having risk and performance transparency.

Figure 8 shows the most commonly monitored security events within the testing interval. Oracle latency was the most frequent single event, fuelled by gas fees during network congestion. Reverts of smart contracts, incorrect wallet access, and API delays also formed a small percentage of events which from a risk management standpoint needed to be addressed.

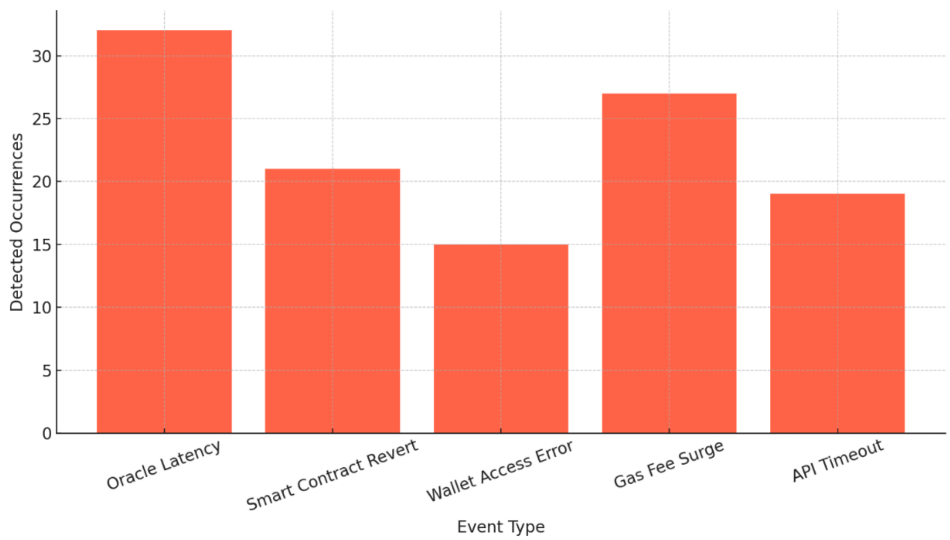


Figure 8: Frequency of Security Events Detected by Type

Automated detection and dynamic control systems act as key components of any system. Functioning safeguards such as fail-safes, fallback routing, and visual supervision dashboards which assist in automation balanced with strong governance, spans enterprise-grade payment platforms.

### 6.3 KYC/AML Compatibility and Logging

One major challenge that comes from DeFi technology, despite its many advantages, is its compliance with rules such as KYC (Know Your Customer) and AML (Anti-Money Laundering). These have-set procedures exist within centralized banking systems, but the decentralized shielded identity aspect of DeFi was largely considered incompatible with any form of id regulatory oversight. This study proves that within DeFi-integrated SAP environments, KYC and AML compliance can be maintained without compromising either system.

In the architectural solutions, KYC was performed at the vendor onboarding stage within SAP. Each vendor record was updated to include wallet addresses and was authenticated through external identity providers vis a vis SAP CPI. The system interfaced with Trulioo type systems that verified documents, tax identification numbers, and business registrations instantly. Only those vendors who passed these checks were permitted to transact using the DeFi payment pathway. In the case that a wallet address did not pass verification or was determined to be associated with a flagged jurisdiction, SAP would automatically halt DeFi payment initiation and instead route the invoice for additional review.

The enforcement of AML was applied at the transaction level. Each outgoing payment via smart contract Triggered an oracle verification against watchlists. The oracle checked real time databases against OFAC and UN sanctioned countries and returned a compliance flag. In cases where the wallet address was flagged, execution of the smart contract was disabled and event logged by SAP as a regulatory breach attempt. These checks functioned alongside SAP's Treasury Risk Management system, where previously consolidated AML scores and country risk monitoring were already in calculation processes.

Monitoring logging enabled all security and compliance workflows. Every event, including transactions that were successful, failed, or reverted, was logged with a timestamp, wallet metadata, transaction hash, SAP object reference, and status codes from both CPI and the blockchain. These logs were captured within SAP Information Lifecycle Management ILM and Document Management System DMS repositories. The data was encrypted, kept per relevant retention policies, and was coordinated with SAP internal audit workflows enabling pre-defined review processes.

In this way, total operational visibility with no regard to supervisory intervention for all DeFi payment activities was achieved, together with operational automation and compliance. Compliance conditions could be verified without manual intervention or payment delays.

# 7 Stakeholder Experience and System Usability

### 7.1 User Feedback from Finance and Compliance Teams

No matter how sophisticated an enterprise system is, its integration requires acceptance from the organization's members. For assessing the business impact of the DeFi-SAP integration, feedback was gathered from key stakeholders in the finance, compliance, procurement, treasury, and IT departments. Participants took part in several simulation sessions in the DeFi-enabled SAP environment and subsequently evaluated the system on parameters of speed, transparency, ease of use, and trust in the compliance mechanisms.

The finance team has reported satisfaction over the reduced processing times and confirmation dashboards. Accounts payable managers noted how payment finality visible within minutes and smart contract logs greatly improved traceability compared to batch payment systems. Treasury users praised the real-time visibility of gas fees and FX conversion as well as their ability to monitor these directly within SAP.

Compliance teams were sceptical at first, especially regarding wallet anonymity and jurisdictional enforcement. However, integration with KYC checks and dynamic sanction screening improved trust significantly. Many compliance officers pointed out that audit reporting was much easier because of the immutable nature of blockchain records and the automated transaction hash synchronization into SAP logs.

Figure 9 summarizes user satisfaction scores for all departments. The treasury department reported the highest satisfaction score, followed by finance and compliance. Marks from the IT department were somewhat lower, pointing to the need for more instruction on Web3 tools and the required middleware settings for initial automation. Satisfaction levels, however, were high across the board which indicates the focus of the system's design on ease of use, transparency, and control was indeed effective.

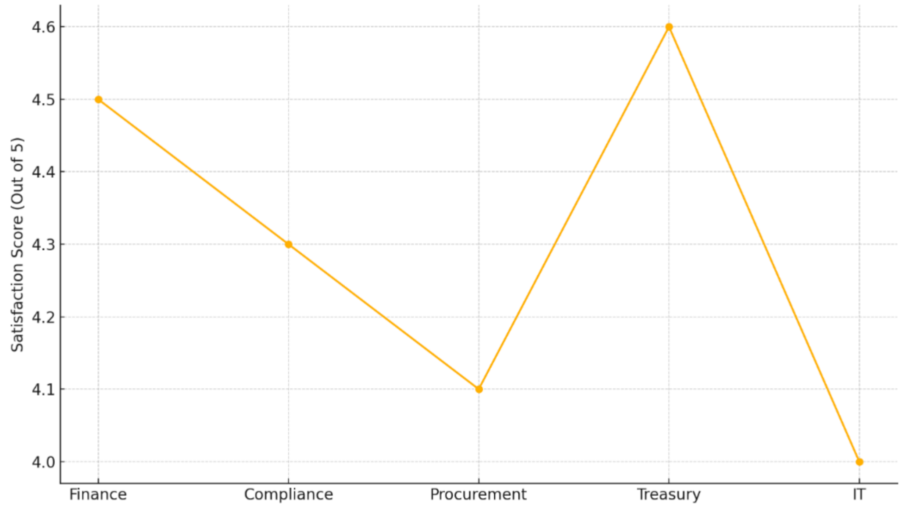


Figure 9: User Satisfaction Scores Across SAP Departments

### 7.2 Usability of DeFi Interfaces via SAP Fiori

One of the main goals of the integration project was to guarantee that business users could interact with and manage DeFi functionalities through standard SAP interfaces. To this end, new tiles and views were added to SAP Fiori applications that enable users to initiate, monitor, and verify DeFi-based payments while remaining in the ERP system.

Fiori's interface was meant to simplify blockchain complexities. Users could approve vendor invoices and authorize payments like they would in SAP, while smart contracts, oracle calls, and gas payments were taken care of behind the scenes. SAP’s document flow was also integrated as users received blockchain transaction hashes, wallet balances, and real-time updates on payment confirmations.

Preliminary responses from procurement and finance roles highlighted an unexpected wallet address, gas fee, and oracle source explanation delay, which seemed to hinder adoption at first. Despite that, after a brief onboarding session, the vast majority said they felt very confident using the DeFi dashboards. Usability within Fiori was enhanced considerably by the addition of tooltips, help tiles, and quick guides.

In addition, audit functionality was incorporated without any disruption in workflow. Retrieving the entire payment history, starting with invoice posting and ending with blockchain confirmation, showed auditors enhanced transparency and diminished traceability burdens with IT and compliance. Visual logs and timeline views provided by the audit feature were deemed more useful by finance managers than the lengthy statement reconciliations.

The success of the Fiori interface serves as a case study of experience design in the context of blockchain adoption. The project de-risked and abstracted many underlying complexities by providing a user interface that non-technical personnel could easily interact with, thereby enabling the DeFi system to be operational.

### 7.3 Organizational Readiness for DeFi Adoption

Alongside the technical aspects of feasibility and ease of use, there exists a more fundamental question concerning an enterprise’s readiness: whether the DeFi integration can be supported and expanded upon. In this regard, adoption readiness scores were calculated across departmental roles to evaluate perception of usefulness, trust in the system, and the level of willingness to surrender conventional processes.

Figure 10 depicts the readiness scores across various departmental roles. Treasury analysts were the most optimistic, expressing strong agreement with the positions afforded by DeFi technology, especially the real-time FX transaction and liquidity management. Finance managers also scored high, motivated by improved operational efficiency and reduced costs. Initially reluctant, compliance officers became much more ready after some exposure to the integrated AML/KYC controls.

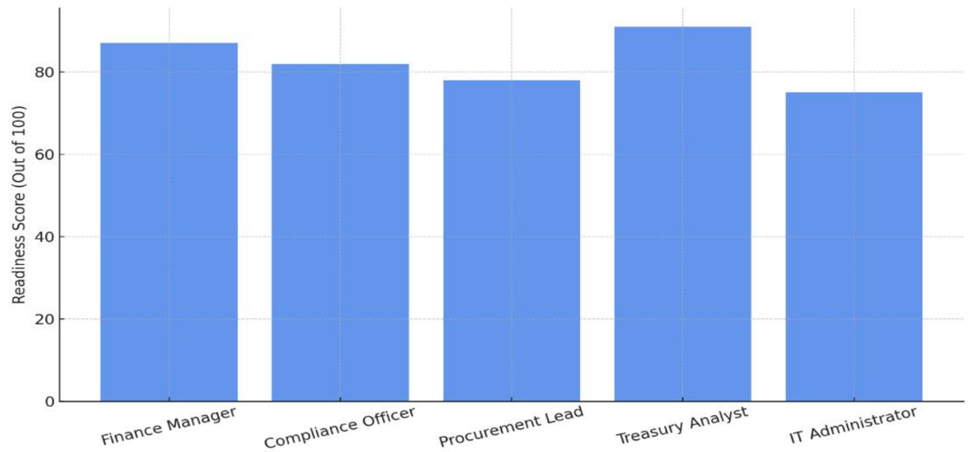


Figure 10: Adoption Readiness by Departmental Role

The readiness scores for procurement leads and IT administrators were marginally lower. Procurement users required further training specific to wallet onboarding and contract payment processes. System maintenance, data recovery, and the overhead associated with supporting external APIs and Web3 modules were concerns raised by IT administrators. Nonetheless, both groups appreciated the value automation offered in the long run and expressed interest in greater participation during future rollouts.

The findings indicate that initial scepticism, especially in the blockchain-averse departments, tends to fade with time when structured onboarding and governance interfaces are provided. For ease of transition, change management activities, multi-discipline workshop sessions, and risk assurance approaches should be bundled with any large-scale DeFi implementation as part of a single offering.

The overarching stakeholder feedback validates that DeFi integration within SAP is not only feasible from a technical standpoint, but is also reasonable organizationally. Adjusted governance workflows and compliance frameworks alongside purposeful change management unlocks growing enterprise readiness to incorporate decentralized infrastructure as a foundational component of the financial architecture.

# 8 Conclusion and Future Scope

## 8.1 Summary of Key Findings

This research showed that the incorporation of DeFi protocols into SAP ERP systems transforms payment automation for enterprises. The integration of smart contracts on the blockchain with conventional SAP finance modules was found to greatly improve the speed, transparency, and cost-effectiveness of cross-border transactions. For a range of scenarios such as vendor payments, recurring disbursements, and intercompany transfers, the DeFi-enabled workflows achieved superior results in both operational and financial metrics compared to conventional workflows. The system attained finality in seconds, reduced costs by over 90%, and enabled real-time traceability through Fiori dashboards and on-chain log retrievals. Received stakeholder input from finance, compliance, treasury, and IT functions validated the benefits and practicality of the integration, confirming its value in large-scale enterprises.

### 8.2 Limitations and Challenges

While the system has accomplished a lot, there are still issues that need to be resolved before it can be widely used. For IT managers, particularly in companies that lack familiarity with Web3 paradigms, the middleware setup and management of blockchain APIs incur an additional layer of technical difficulty. Smart contracts executed in diverse DeFi ecosystems incur compatibility, latency, and oracle dependency risks. Compliance integration is valid but relies on KYC providers, and the region-dependent sanction feeds may differ greatly. In addition, the system's dependence on the Layer 1 and Layer 2 networks exposes it to congestion and volatility, impacting the timeliness or cost of transactions intermittently. From the system's side, onboarding requires active retraining and change management, particularly in procurement and compliance functions that are not yet blockchain-friendly.

### 8.3 Future Work: DeFi Lending, Staking, and CBDC Interoperability

Spanning the architecture to accommodate more sophisticated DeFi functions like lending, staking, and real-time liquidity provisioning can be done latterly. With the incorporation of decentralized lending protocols, SAP Treasury functions could automatically control working capital or directly borrow from DeFi markets against receivables. Staking could be economically automated to yield returns for the idle treasury funds within rules-based SAP workflows. Another critical avenue of interest is the interoperability of DeFi protocols with Central Bank Digital Currencies (CBDCs). With changing norms, the ability for SAP systems to interface with CBDCs through hybrid smart contracts would revolutionize enterprise digital finance. These enhancements would not only broaden and deepen financial features of the system, but also catalyse the positioning of SAP ERP at the core of the shift towards programmable and decentralized economies.

## References

[1] Almaliki, O. J., & Al-saedi, M. O. (2023). The Impact of the Qualitative Peculiarities of Accounting Information Based on the Financial Reports of Commercial Banks. *International Academic Journal of Social Sciences, 10*(1), 49–56. https://doi.org/10.9756/IAJSS/V10I1/IAJSS1006

[2] Campos, R. B. (2024). The Impact of Digitalization on Credit Risk Management in Microfinance Institutions in Nueva Ecija, Philippines. *Indian Journal of Information Sources and Services, 14*(3), 145–156. https://doi.org/10.51983/ijiss-2024.14.3.20

[3] Deutsche Bank, (2022). Cross-border payments: A roadmap to a frictionless future. *Deutsche Bank Corporate Banking*. https://corporates.db.com/files/documents/publications/DB-G20-Roapmap-WP-40pp-Web-Secured.pdf

[4] Duan, C., Soh, W. N., Ong, T. S., & Rahim, N. A. (2024). Strategies for Safeguarding Financial Stability: Integrating Government Intervention, Financial Freedom, and Information Management Systems for Resilient Economies. *Journal of Internet Services and Information Security, 14*(4), 181-194. https://doi.org/10.58346/JISIS.2024.I4.010

[5] Duvey, A. A., Goyal, D., & Hemrajani, N. (2012). A reliable ATM protocol and comparative analysis on various parameters with other ATM protocols. *International Journal of Communication and Computer Technologies, 1*(1), 28-33. https://doi.org/10.31838/IJCCTS/01.01.03

[6] Ghazanfari, M., Ramezan, M., & Vanda, M. (2018). The impact of motivation on commitment through private bank branches employees with mediating role of identity style. *International Academic Journal of Organizational Behavior and Human Resource Management, 5*(1), 84–94. https://doi.org/10.9756/IAJOBHRM/V5I1/1810005

[7] Glaser, F., Hawlitschek, F., & Notheisen, B. (2018). Blockchain as a Platform. In *Business Transformation through Blockchain: Volume I* (pp. 121-143). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-98911-2_4

[8] Jamithireddy, N. S. (2024). Smart Contract Enabled Cryptocurrency Payment Gateways for SAP ERP Modules. *Research Briefs on Information and Communication Technology Evolution*, *10*, 231-251. https://doi.org/10.69978/rebicte.v10i.208

[9] Kamble, S. S., Gunasekaran, A., & Sharma, R. (2020). Modeling the blockchain enabled traceability in agriculture supply chain. *International journal of information management*, *52*, 101967.

[10] Laghareh, F. S., Mirabedini, S. J., & Abadi, A. H. (2018). Provide a method for validation of bank customers using data mining techniques (Case study: Bank Sepah). *International Academic Journal of Science and Engineering, 5*(2), 145–153. https://doi.org/10.9756/IAJSE/V5I1/1810032

[11] Liu, C., Bodorik, P., & Jutla, D. (2024). Automated Mechanism to Support Trade Transactions in Smart Contracts. *Revision submitted for publication]. In Journal of Blockchain: Research and Applications*.

[12] Luu, L., Chu, D. H., Olickel, H., Saxena, P., & Hobor, A. (2016, October). Making smart contracts smarter. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security* (pp. 254-269). https://doi.org/10.1145/2976749.2978309

[13] Nadir, R. M. (2019, November). Comparative study of permissioned blockchain solutions for enterprises. In *2019 international conference on innovative computing (ICIC)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICIC48496.2019.8966735

[14] Saberi, S., Kouhizadeh, M., Sarkis, J., & Shen, L. (2019). Blockchain technology and its relationships to sustainable supply chain management. *International journal of production research*, *57*(7), 2117-2135. https://doi.org/10.1080/00207543.2018.1533261

[15] Scharfman, J. (2021). Decentralized finance (DeFi) compliance and operations. In *Cryptocurrency Compliance and Operations: Digital Assets, Blockchain and DeFi* (pp. 171-186). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-88000-2_9

[16] Singh, D. S. (2024). Decentralized finance (DeFi): Exploring the role of blockchain and cryptocurrency in financial ecosystems. *International research journal of modernization in engineering technology and science*, *5*. https://www.doi.org/10.56726/IRJMETS48585

[17] Torres, C. F., & Steichen, M. (2019). The art of the scam: Demystifying honeypots in ethereum smart contracts. In *28th USENIX Security Symposium (USENIX Security 19)* (pp. 1591-1607).

[18] Treiblmaier, H. (2018). The impact of the blockchain on the supply chain: a theory-based research framework and a call for action. *Supply chain management: an international journal*, *23*(6), 545-559. https://doi.org/10.1108/SCM-01-2018-0029

[19] Zhang, L., Zhang, Z., Wang, W., Jin, Z., Su, Y., & Chen, H. (2021). Research on a covert communication model realized by using smart contracts in blockchain environment. *IEEE Systems Journal*, *16*(2), 2822-2833. https://doi.org/10.1109/JSYST.2021.3057333

## Author Biography

**Naren Swamy Jamithireddy** received his Masters degree in Information Technology and Management from Jindal School of Management, The University of Texas at Dallas, USA in 2014. Currently, he is working as an Advisory Manager at Deloitte & Touche LLP, pursuing research in Enterprise Resource Planning (ERP) and Generative AI.