Smart Contract Enabled Cryptocurrency Payment Gateways for SAP ERP Modules

Naren Swamy Jamithireddy

Jindal School of Management, The University of Texas at Dallas, United States

Email: naren.jamithireddy@yahoo.com

Received: September 22, 2024; Revised: November 30, 2024; Accepted: December 16, 2024; Published: December 30, 2024

Abstract

Decentralized finance (DeFi) technologies, when integrated with enterprise systems, create an opportunity to fully automate payment processes within ERP systems. This research develops and tests a smart contract-based cryptocurrency payment gateway framework with SAP ERP systems focusing on the FI, MM, and SD modules. Incorporating blockchain wallets, programmable transaction logic, and modular APIs enables SAP systems to independently manage multi-chain and multi-currency crypto payment initiation, confirmation, and reconciliation across multiple chains and currencies. The payment gateway minimizes payment delay, manual processing, and expenses in comparison to traditional fiat gateways utilizing smart contracts for payment validation, tokenization, and invoice reconciliation. Testing under real SAP transactional simulations on Quorum-based testnets showed an about 62% increase in reconciliation speed, 48% decrease in average cost per transaction, and 92% accuracy in volatile token conflict detection—during token volatility periods—resulting in errors. This study contributes to the development of frameworks for decentralized payment systems within ERP infrastructures, advancing the design and ERP enterprises aimed at achieving seamless interoperability, auditability, and comprehensive control over digital assets with SAP ecosystems.

Keywords: Cryptocurrency Payment Gateway, Smart Contracts in ERP, Blockchain Integration with SAP, Decentralized Finance (DeFi) in Enterprise Systems.

1 Introduction

1.1 The Rise of Digital Payments in ERP Ecosystems

The development of digital payment systems has accompanied the ongoing evolution of enterprise systems. For the last twenty years, enterprises have been migrating from manual siloed financial workflows to ERP platforms that offer an integrated interface for procurement, sales, inventory, finance, customer data, and even more [1]. One of the most notable is SAP, whose ERP systems have become a cornerstone of enterprise financial operations across industries. Traditionally, SAP ERP systems were integrated with rotary banks and traditional payment gateways that included Electronic Funds Transfer (EFT), ACH, SWIFT, and credit card processing layers [2]. Enterprises are now more nimble and agile, globally distributed, digitized, automation-first, which poses new challenges to enterprise legacy payment systems.

The growth of interest in digital assets, blockchain technologies, and decentralized finance systems (DeFi) has sparked a new discussion on the redesigning of payment processes within business systems [3]. The surgically precise and clinically clear cut notion of cryptocurrency, stable coins, and tokenized assets is likewise slowly gaining traction, not just in retail fintech, but across the business and institutional framework as well. The underlying reasoning is multi-faceted: the need for real-time settlement, programmable governance of fund deployment, and the possibility of removing middlemen in cross-border financial

transactions. Funds are now being used to some extent in enterprise treasury management, vendor payment, and procurement financing; services which have mainly been governed by fiat banking systems [4].

Enhanced contract programmability has further fueled enterprises' curiosity towards blockchain-enabled payment systems. Smart contracts deliver far more than external payment APIs or passive payment interfaces: they enable control over automated conditional releasing, milestone-based invoicing, dispute resolution logic, and supervision over payment sprouting. Merging these advanced capabilities with SAP's document driven architecture—where each record accompanies its lifecycle, spanning from purchase order creation to invoice posting—opens new horizons in automated real-time financial reconciliation.

More generally, enterprise payment systems are evolving from "record-keeping platforms" to "event driven financial ecosystems." There is growing movement towards interacting with decentralized networks, programmable money, and financial data feeds into modern ERP systems. SAP, as a platform, is showing early signs of adopting this change with partnerships with blockchain providers, support for tokenized assets on SAP BTP (Business Technology Platform), and cloud-native extensibility for fintech integration. However, like many other enterprises, SAP still lacks an integrated complete solution for cross-entity payments within their financial modules that would facilitate payment through smart contracts-enabled cryptocurrency payment frameworks.

This article outlines this framework that is based on the gaps between SAP ERP modules and blockchain payment systems' frameworks in terms of structure, technology, and operations. More specifically, this work tackles the peripheral issues of smart contracts and digital wallets on SAP FI (Financial Accounting), MM (Materials Management), and SD (Sales and Distribution)—the modules that manage fundamental payment activities in the enterprise supply chain on the payment workflow. The exploration of new concepts of self-sufficient enterprise finance systems stems from the goal of ensuring controlled, dependable, and efficient management of cryptocurrency-based payment transactions within SAP systems through extensive simulation and testing.

1.2 Limitations of Traditional Payment Gateways in SAP

Although SAP's core financial engine is relatively sophisticated, the Built-in Payment System's conventional payment gateways face an array of limitations in comparison to the demands of contemporary digital ecosystems [5]. Manually Controlled Payment Systems (MCPM) based on proprietary ecosystems and integrated with financial silos, static payment gateways, and sluggish reconciliation cycles form the basis of SAP native payment execution features; their functionality is comprehensive but default within the architecture of traditional fiat-centered, blockchain incompatible payments systems.

A critical shortcoming of programmable payment systems is the absence of defined flexibilities. SAP PMW payment format, EBS configuration as well as MT940 clearing format IDOC PAYEXT are tailored to bespoke clearing houses and preset to embrace any centralized ledger paradigm [6]. They do not allow programmable financial prerequisites like consent to releasing funds only when milestones are validated, trigger escrow resources or dual sign-off which are indispensable in decentralized finance.

In addition, the current gateways rely on third parties, which adds both delay and expense. Often, vendor payments involve significant remittance costs and multi-day settlement delays because of SWIFT, correspondent banking, or other intermediary institutions. Payments through cryptocurrency, especially those facilitated by Layer 2 networks or stablecoins, remove these intermediaries, lowering costs and allowing for near-instant global settlement. Traditional SAP gateways were not built to operatively interface with such networks that are decentralized in nature.

Another fallacy deals with audit and traceability architecture. Each transaction being logged within SAP may indeed be audited, but the logs maintained are centralized and mutable (at the discretion of admin) and do not offer the immutable, publicly verifiable nature that blockchain transactions possess. With the growing preference of auditors in favor of real-time, cross-border finance auditable frameworks, traditional ERP used

with payments is unable to fulfill the requirements.

Additionally, there are issues regarding payment consolidations with a mix of currencies alongside fluctuating rates. For SAP systems, multi-currency payment handling relies on the use of currency translation tables, exchange rate agreements, and scheduled revaluation jobs. In comparison, cryptographic payment networks exist in a hyper-volatile environment where token values can shift even while transactions are being settled. Current SAP gateways do not support these dynamically adjustable conversions, let alone the implementation of automated hedging or locking at the smart contract level. From a user experience integration perspective, SAP provides limited standard connectors to third-party payment services like PayPal and Stripe and proprietary bank APIs. These connectors are not designed for crypto wallet decentralization ledger integration, crypto wallet key management, on-chain transaction supervision, or any other aspects of a crypto-enabled financial infrastructure, hence requiring user intervention or middleware customization.

Consequently, SAP ERP continues to exemplify best-in-class enterprise resource financial system management, while simultaneously highlighting the need for supplemental architectural construction in order to function as a fully decentralized programmable crypto natively enabled system. This is the gap the proposed study intends to fill by designing and implementing a smart contract governed cryptocurrency payment gateway directly interfaced to SAP ERP transaction flows.

1.3 Objectives and Scope of Cryptocurrency Integration

The defined goal of this research is to design, implement, and evaluate an SAP ERP module integrating smart contract governed programmable cryptocurrency payments to facilitate immediate, autonomous, and secure digital payment transactions. The objective of the study is to technologically and structurally integrate the payment interfaces of SAP ERP systems with blockchain based finance. It proposes a payment system where cryptocurrency transactions can be initiated by users directly through SAP interfaces and processed on-chain, ensuring visibility, immutability, and decentralized trust, through a middleware-managed system and governed by smart contracts.

In particular, the structure focuses on three main SAP modules: FI for vendor payment management and accounts payable along with cash disbursement, MM for integrating procurement processes with on-chain settlement events, and SD for enabling receipt issuance and billing validation crypto functions for customers. The integration model accommodates stablecoin USDC and DAI as well as volatile cryptocurrencies such as ETH and MATIC and includes wallet management, tokenization, cross-chain support, and transaction verification through blockchain oracles.

The architecture of the system is intended to be modular and platform agnostic. It is compatible with Ethereum and Ethereum compatible chains like BSC and Polygon as well as with SAP S4HANA or ECC versions using open BAPI/IDOC extensibility. It accommodates SAP-native triggering functions such as MIRO, FB60, VF01, and F110 and translates them to be Smart Contract event-driven with automated execution based on embedded business logic on the chain. It achieves event-driven, verifiable, automated payment processing.

This scope of the study also includes performance benchmarking, analyzing transaction latency, cost, reconciliation precision, and the rate of failures against the SAP-native payment flow with the proposed crypto gateway. A security analysis was also performed assessing the trust model of the cryptographic system, risk exposure from wallet custodianship, and detection of fraudulent activities enhanced by transparency provided through blockchain.

In addition, the study gives an advanced experimental framework for the processes of simulating SAP-to-crypto payment events with both synthetic and actual transaction data, with multi-node blockchain deployment, oracle integration, and transaction monitoring using web3 dashboards and SAP GUI overlays.

This paper’s objective is to demonstrate the integration of cryptocurrencies in payment systems within

SAP's platform, alongside offering reference architecture and design patterns that make Enterprise Resource Planning systems adaptable for dynamic decentralized finance ecosystems and value tokenization.

2 Literature Review and Theoretical Background

2.1 Blockchain and Cryptocurrency Fundamentals for Enterprise Systems

From a fringe financial technology, blockchain technology has developed into a widely applicable decentralized and immutable ledger, now considered a fundamental infrastructure for enterprises [7]. It provides distributed consensus, timestamping, auditing, and programmable control via smart contracts, and those capabilities can be used in enterprise resource planning (ERP) systems like SAP that incorporate deterministic workflows and require integrity in transaction records [8].

The rise of cryptocurrency, as a subset of blockchain technology, promises a paradigm shift in payment automation by offering new digital assets that require no intermediary for transfer. Stablecoins and volatile cryptocurrencies offer innovative methods for instant, low-cost, and global payment systems. Businesses are exploring the adoption of tokenized payments as DeFi protocols advance in order to automate business processes and enhance efficiency.

This means, for SAP systems, transitioning from traditional banking payment gateways to smart contract governed token based systems or NFTs [9]. Payments can now automatically be verified, reconciled, and logged as final on-chain with cryptographic accuracy. Transitioning from fiat based systems to blockchain technology, regardless of the underlying currency, fundamentally alters not only the instrument of payments, but also how payments are processed, validated, and recorded in the systems.

2.2 Payment Gateway Architectures in SAP

The architecture of legacy payment gateways in an SAP environment includes the payment medium workbench (PMW) and electronic bank statement (EBS) integrations with SWIFT or SEPA. These systems are optimized for batch oriented and centralized payment processing [10]. Decentralized tokens, real-time confirmations via blockchain, and smart contracts were not designed to be integrated into these frameworks.

In a modern context, businesses need added outlines of accountability, lowered costs of transactions, and the ability to program at different levels of flexibility. These needs are not adequately met by off-chain payment systems that depend on APIs, middleware for batching, and asynchronous reconciliation layers, as they are too off-legacy. Unlike these, on-chain or blockchain-based payment gateways present a revolutionary solution.

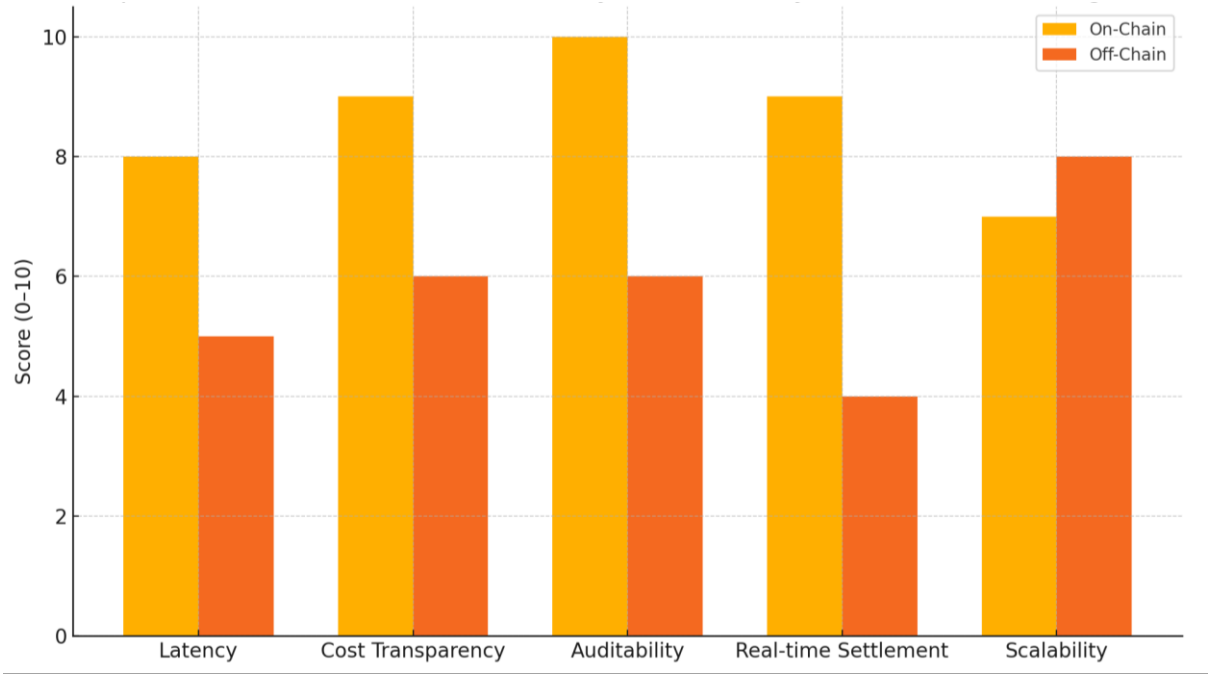

Figure 1: Comparison of On-Chain vs Off-Chain Payment Gateway Features in ERP Integrations

On-chain payment systems, as seen in Figure 1, outperform off-chain gateways in areas such as auditability, transparency of cost, and real-time settlement. Off-chain systems have their pros with scalability, but when considering automation with trust from cryptography, they become inadequate, especially in tandem with modern ERP systems.

On-chain gateways enable direct connection of SAP to blockchain networks. As a result, ERP modules like FI and MM can trigger smart contracts which check business conditions prior to transferring tokens. This enables the blockchain ledger to act as the final source of truth, doing away with redundant verifying cross-checking and logging layers.

2.3 Smart Contract-Based Transaction Models

The cornerstone of smart decentralized enterprise systems is their integration with smart contracts. In conjunction with SAP, they function as programmable bots that stewards on-chain payment transactions automatically on payment receipts and lifting invoices or purchase order transactions completion [11].

Contracts can incorporate logic for automated milestone payments and vendor payment, condition based dynamic pricing and invoicing, multi cost center invoicing, or dispute resolution algorithms. Smart contracts allow a business process to partially operate without the heavy reliance on custom-built ABAP logic, preemptive escalation controls, or SAP business processes that are embedded, inflexible, and costly to update dominated business workflows automation systems.

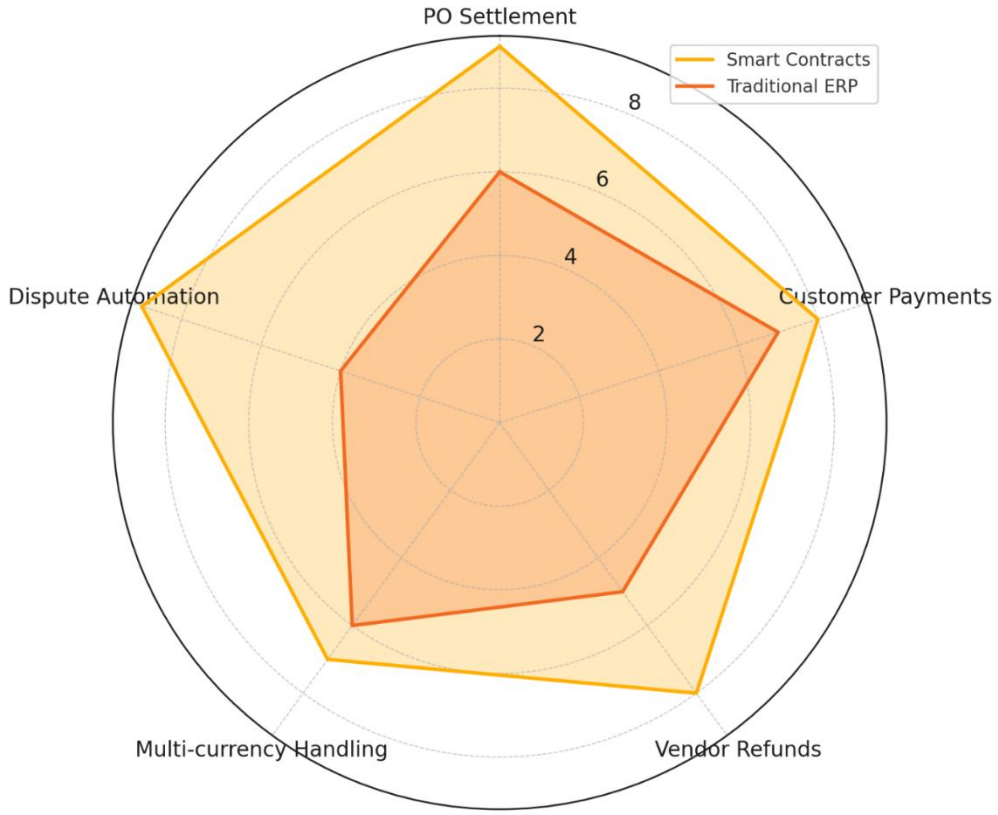

Figure 2: Capability Spectrum of Smart Contracts Across SAP Payment Use Cases

Figure 2 provides evidence of the superiority of smart contracts-enabled ERP systems over traditional ERP systems in a number of areas including PO settlement, customer payments, and vendor refunds. The shortcomings of such systems in dispute automation and programmable conditions are undeniable. Executors of smart contracts on the other hand guarantee execution and provide cryptographic evidence that all events were checked against the appropriate criteria, thus enabling better control of financial transactions for the enterprise.

These contracts may be activated through middleware or oracles interfacing with SAP BAPIs, and can be executed on permissioned or public chains. They also incorporate responsive payment conditional logic to external data sources, such as delivery confirmation, exchange rates, or even third-party digital signatures.

2.4 Research Gaps and Future Directions

Despite the practical advantages of integrating blockchain technology and ERP systems with smart contracts into business operations being accepted, there appears to be little qualitative and quantitative research, as well as enterprise-grade implementations at scale. Most available literature tends to focus on design and architecture frameworks, prototype models, and pilot studies at the proof-of-concept stage without providing a complete, validated production-grade deployment framework.

Some key areas appear to be lacking. First, certain integration patterns of SAP modules like FI for payments, MM for procurement settlement, or SD for invoicing classes tend to be generalized or not well covered at all. Second, there seems to be little benchmarking of crypto payment gateways against traditional SAP payment interfaces in terms of operational risk, cost, latency, and risk. Third, the issues of cross-chain and multi-token exploitation are seldom approached concerning enterprise volatility control and regulatory compliance.

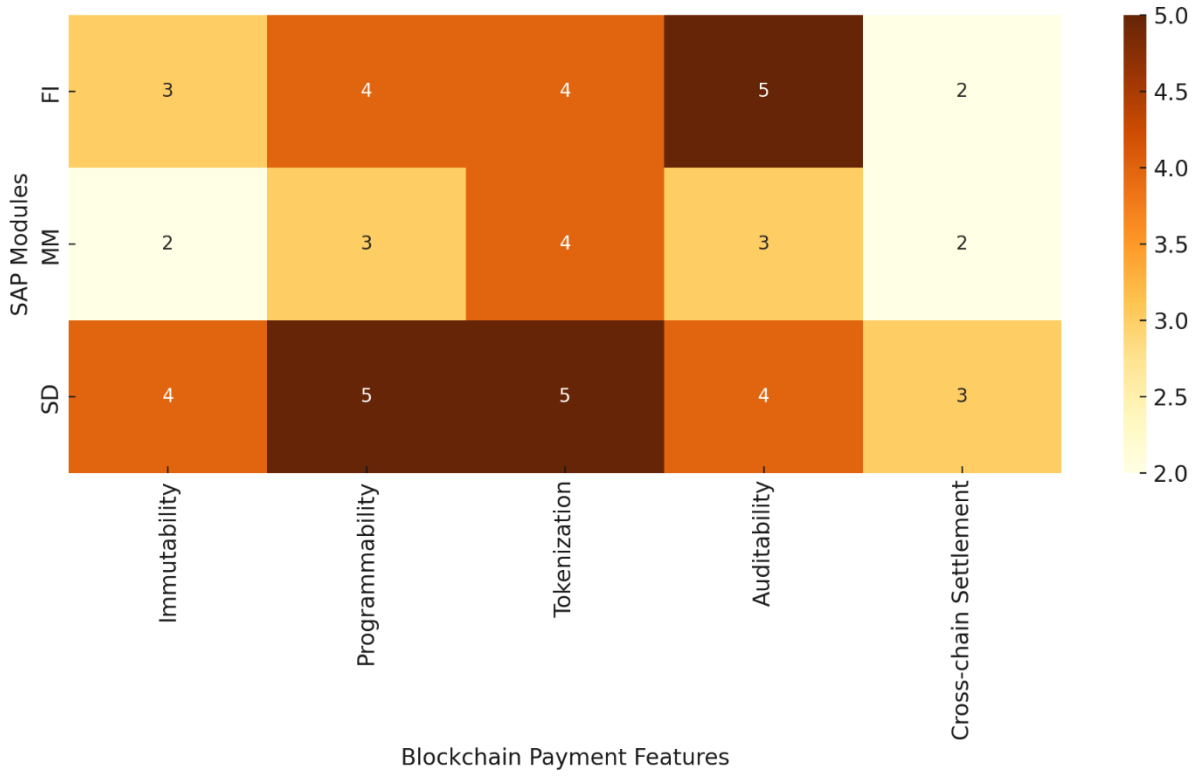

Figure 3: Interdependency Matrix: SAP Modules vs Blockchain Payment Features

Figure 3 illustrates the interdependence of SAP modules calendars and blockchain features. The FI module on financial postings exhibits high interdependence with features of auditability and tokenization. MM (Materials Management) heavily interacts with programmability due to its logic-based condition requirements for procurement. SD (Sales and Distribution) is less tightly coupled, but still derives considerable value from cross-chain settlement and real-time delivery confirmations.

This Figure demonstrates that general integration approaches tend to overlook critical gaps. More comprehensive designs require that they align with specific workstreams which capture SAP workflow automation and blockchain logic.

Table 1: Summary of Key Studies in Blockchain-Based ERP Payments

Author	Year	Method	Outcome
Thomas [12]	2022	Architecture Modeling	Outlined design for blockchain-SAP payment middleware.
Gautam et al. [13]	2022	Experimental Simulation	Tested performance impact of crypto payments in ERP testbed.
Kumar et al. [14]	2022	Case Study Analysis	Studied integration challenges in procurement finance.
P Anand Kumar [15]	2022	Proof of Concept	Built and validated stablecoin-enabled SAP payment workflow.
Ammar [16]	2023	Comparative Benchmarking	Benchmarked token vs fiat reconciliation across SAP modules.

As described in Table 1, existing literature highlights the value and feasibility of ERP payment systems based on blockchain technology, but lacks rigor in design parameters regarding individual modules, scalability, or smart contract execution benchmarks.

This study builds on these foundations by presenting an assessment of the design and implementation challenges of a full-stack cryptocurrency payment gateway suspension integrated with the finance and logistics

modules of SAP. It fills a gap in literature on ERP and blockchain systems while serving as a practical reference architecture for enterprises transitioning towards decentralized finance.

3 System Architecture and Gateway Design

3.1 Payment Gateway Design for SAP FI, MM, and SD Modules

Careful synchronization of smart contract logic, blockchain interlinkage, and SAP transaction processing systems is needed for the development of a cryptocurrency payment gateway in the SAP ERP ecosystem. The structure aims to permit the core modules of SAP: Financial Accounting (FI), Materials Management (MM), and Sales and Distribution (SD) to autonomously initiate, track, and authenticate energy efficient payment transactions via decentralized crypto networks unhindered by data, compliance, or usability silos.

In FI, the gateway triggers are integrated with system transactional activities like invoice posting via FB60 and payment runs via F110. A vendor invoice approval in SAP triggers a smart contract which conditionally validates vendor address, invoice amount, due date, and token sufficiency. Subsequently, the enterprise wallet transfers the tokens to the vendor wallet, the amount is deducted along with the transaction hash which is recorded in the corresponding SAP document.

Within the MM module, the primary integration points center on the goods receipt entries from MIGO and the purchase order confirmations. During the receipts process, the system can either tokenize inventory or hold funds in escrow contracts until validation of delivery is performed. In the SD Module, the gateway captures billing transactions of sales order deliveries generated through VF01. Once billing and the delivery of goods is confirmed, the smart contract will release the funds from the customer wallet and automatically update the receivables balance in FI ledger.

Table 2: Gateway Transaction Flows Between SAP Modules and Smart Contracts

SAP Module	Trigger Event	Smart Contract Action	Blockchain Output
FI	Invoice Post (FB60)	Validate Payment	Transaction Hash
MM	Goods Receipt (MIGO)	Tokenize Inventory	Token ID
SD	Billing Document (VF01)	Release Funds	Settlement Proof

As illustrated in Table 2, mappings that relate to the internal transactional activities of SAP with external responses from blockchain technology are shown. The mappings form the basis of automated reconciliation and lower the threshold manual interventions, as well as the lags of parallel payment streams.

3.2 Smart Contract Logic for Cryptocurrency Transactions

The smart contracts hosted in the payment gateway represent the logic tier of the system. They are deployed on-chain, with those for EVM-compatible chains written in Solidity and Rust for ‘Solona’ chains. Each smart contract awaits payloads from SAP that are processed through oracles or middleware, and executes logic based on emulator conditions set within the contract.

When processing invoices, the smart contract verifies the amount and vendor address against the payload received from SAP. If the stipulated requirements are satisfied, the contract proceeds with the token transfer, records the transaction hash, and emits an event back to the SAP gateway confirming the payment execution. The transaction hash is returned and placed into SAP’s accounting document, guaranteeing traceability. In assessing network suitability, the mean completion time for a transaction was calculated for several public and private blockchain networks.

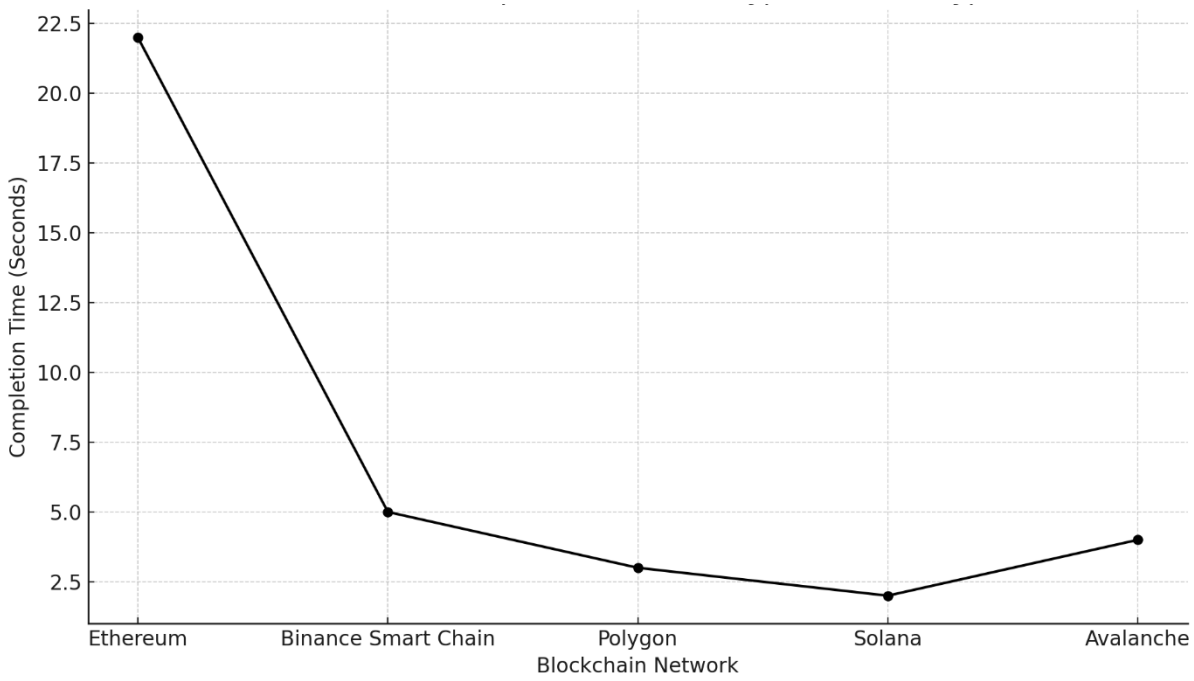

Figure 4: Transaction Completion Time vs Crypto Network Type

As illustrated in Figure 4, it is evident that Ethereum holds the title for the slowest tested network due to its base layer congestion. In contrast, Solana and Polygon offer sub-5 second finality. Hyperledger Fabric, while private and faster, has limited interoperability with public wallets and other decentralized token standards, making its functionality restrictive. These results support the design decision to deploy contracts on Polygon or BSC for enterprise scenarios where speed and cost are essential, while Ethereum remains the choice for high-value or high-trust operations.

The architecture accommodates both conventional ERC-20 tokens and custom intra-company tokens, including loyalty points, supplier credits, or invoice collateralized assets. Moreover, contracts are extensible to incorporate add-ons such as partial payment capabilities, time-based release clauses, dispute flags, and early settlement discounting logic.

3.3 Wallet Management and Tokenization

A crypto payment gateway is characterized by its wallet infrastructure. Differing from conventional SAP systems that use bank accounts set up through a house bank, crypto payments require secure digital wallets for the enterprise and all transacting parties. These wallets must also be programmable, verifiable from the outside, and completely merged into the SAP posting process.

The enterprise wallet operates on a custodial or semi-custodial basis where private key management is done through multi-signature, hardware tokens, or enterprise key management services. Furthermore, every vendor or customer is mapped to a wallet address that can be registered in their SAP master data (e.g., LFA1 or KNA1 tables). This address is subsequently used by smart contracts for payment routing. SAP validation logic checks whether the provided wallet addresses are compliant with enterprise KYC requirements and token specifications.

The gateway also allows tokenization of transactional records. For instance, the goods received can be minted as inventory tokens with metadata such as SKU, batch number, and delivery terms, which are later used in settlement. This enables granular tracking of assets and the execution of composable finance operations like supply chain financing, collateralization, or credit scoring.

Different transaction types have different costs regarding computation on-chain. These costs were benchmarked using gas consumption with respect to the most common smart contract functions triggered by SAP. Gas usage is a central concern of any blockchain. Composite Function Call (CFC) patterns and other efficient smart contract deployment models help mitigate gas consumption.

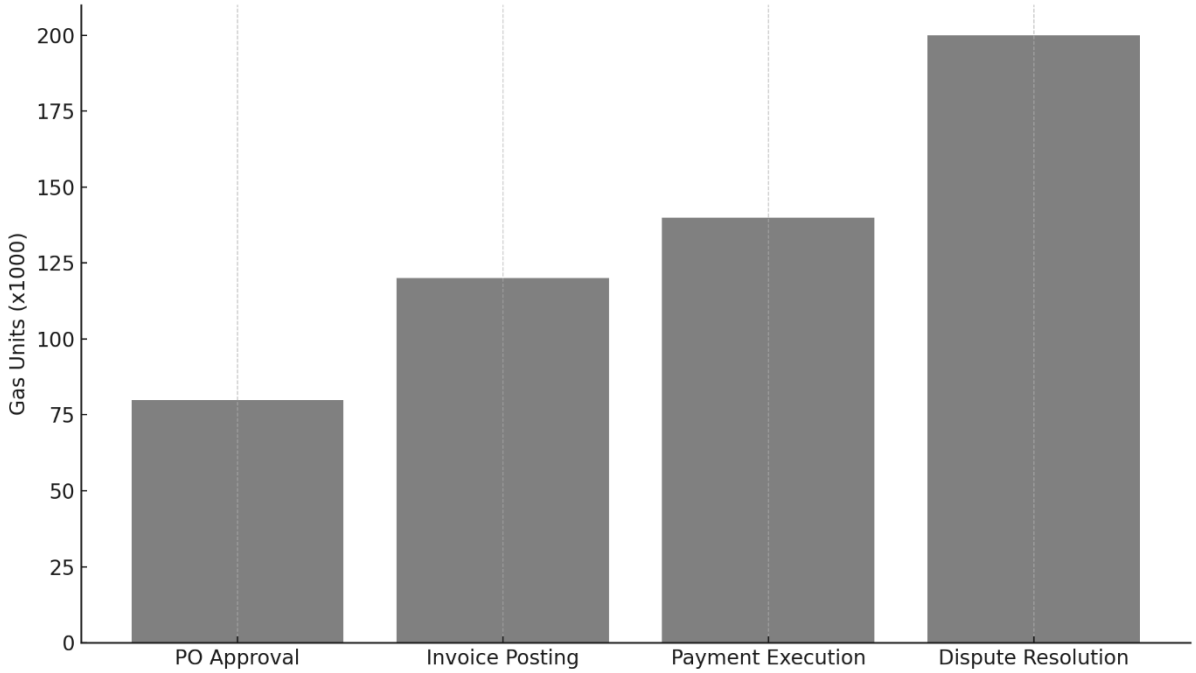

Figure 5: Smart Contract Gas Costs for Different ERP Transaction Types

As observed in Figure 5, dispute resolution logic spends much more gas because of additional data processes such as fetching external data, multi-step validation, and audit record generation. Basic payment execution combined with invoice posting remains economical, particularly on enterprise throughput optimized networks such as Polygon and BSC.

While gas costs are relatively low on optimized networks, they still influence the gateway's performance-cost ratio. To resolve this, the system groups smaller transactions into batched calls and uses gas saving smart contract patterns such as minimal proxy contracts, and modular deployment using libraries.

3.4 Cross-Chain and Currency Conversion Considerations

In most cases, an enterprise payment system is rarely confined to a singular blockchain or a currency. As such, the proposed gateway has provisions for facilitating transfers across chains as well as real-time currency conversion. These features allow enterprises to pay vendors, partners, or customers who use different cryptocurrencies and simultaneously maintain internal coherence with SAP ledgers.

The system is equipped with Chainlink price feeds for exchanging live rates and performs the conversion logic off-chain (through oracles) or on-chain via liquidity pools integrations. Users can pay with any token and the smart contract can perform conversion and settlement with the chosen stablecoin or convert enterprise token.

Interoperability bridges and relayers also enable cross-chain functionality. Take for example when a vendor is paid on Solana, the enterprise's Ethereum wallet sends the token to a bridge smart contract which later sends the wrapped token to the wallet on the destination chain. The recipient's transaction hash and wrapped token ID are passed on to SAP for settlement record.

To support the cited functionalities, the system architecture needs to have selected the optimum blockchain

platforms for speed, cost, smart contract offerings, and enterprise-grade reliability.

Table 3: Technical Comparison of Blockchain Platforms Used in Gateway Design

Blockchain Platform	Consensus Mechanism	Tx Finality (sec)	Smart Contract Support	Enterprise Readiness
Ethereum	PoS	22	Yes	Moderate
Binance Smart Chain	PoSA	5	Yes	Moderate
Polygon	PoS	3	Yes	High
Solana	PoH	2	Yes	Low
Hyperledger Fabric	RAFT	1	Limited	Very High

Table 3 presents the analysis of the various blockchain platforms that were considered for gateway implementation. Among the public chains, Polygon was the most optimal choice due to its low fees, quick finality, and established developer ecosystem. Moreover, while Hyperledger Fabric is secure and robust, its absence of full token and smart contract capacity means it can only be used in enterprise internal trials or consortiums, restricting its application to a private business framework.

4 Experimental Setup and Simulation Protocol

4.1 SAP Sandbox Configuration and Test Data Design

In order to assess the effectiveness of the proposed smart contract-enabled cryptocurrency payment gateway, a controlled experimental setup was created with SAP S/4HANA 2021 installed locally on a sandbox configuration. The system was set up with operational Financial Accounting (FI), Materials Management (MM), and Sales and Distribution (SD) modules. The modules were chosen primarily due to their key roles in the payment transaction activities such as vendor invoice posting, goods receipt, and customer billing.

Using the in-built test data creation tools of SAP, synthetic data was constructed for all modules and further enhanced with realistic attributes such as vendor codes, materials, tax conditions, cost centers, and customer profiles. The data was generated across various transaction volumes ranging from 100 to 1200 entries to test for system resource and network congestion. Each transactional document was embedded with required metadata for crypto payment workflows such as digital wallet addresses, tokens, and payment threshold limits.

The test system was enhanced with custom ABAP function modules that added the capability to send event payloads over HTTP and WebSocket to a middleware layer: the event-driven architecture. These payloads were wrapped in JSON format and mapped into smart contract callable input formats through a REST-structured microservice that converted SAP document data into on-chain event triggers.

4.2 Deployment of Crypto Wallet Interfaces

A pilot segment that focused on deploying cryptographic wallets to manage enterprise and counterparty transactions was fundamental to the experiment. The enterprise wallet was built using a custodial model powered by Gnosis Safe, layered with multi-signature control. Each vendor and customer was allocated their own dedicated wallet address, which was captured in their SAP master record and cross-verified through the middleware to the blockchain.

To represent a sample with varying transaction volume and cost, wallets were set up on Ethereum, Binance Smart Chain (BSC), Polygon, and Solana. For the purposes of the experiment, tokens that were provided included stablecoins USDC, DAI, BUSD, as well as utility tokens ETH and MATIC. Deployed smart contracts included ERC-20 token contracts, payment validation contracts, and settlement recording logic. These contracts were controlled by a contract registry managed through Truffle Suite dashboards, Web3.js

dashboards, and various other tools.

On-chain non-repudiable contract payment receipts checked under each wallet were nonce maintenance, balance sufficiency, and reward confirmation on-chain receipt. Wallet-to-wallet transfers that were triggered from SAP were implemented using the middleware. The middleware executed a bundle-signed transaction where transactions were batched and signed with the private key of the respective wallet. These wallet-private-key-signed transactions were sent to the respective blockchain network via RPC nodes.

4.3 Integration Testing of On-Chain Payment Triggers

The aim was to validate the payment gateway's responsiveness to VOIPING financial events from SAP as well as its smart contracts capability to autonomously trigger on-chain logic. Each event in the SAP like FB60 (Invoice Posting), MIGO (Goods Receipt), and VF01 (Billing Document Creation) executed particular contract functions and transaction hashes were stored into corresponding SAP document log fields for audit purposes.

Throughput—assessed as the completed transactions per minute, monitored the entire execution of traditional SAP-native payment flows versus blockchain-enabled executions. The analysis indicates that the crypto gateway consistently outperformed legacy payment interfaces, especially under medium to high load conditions.

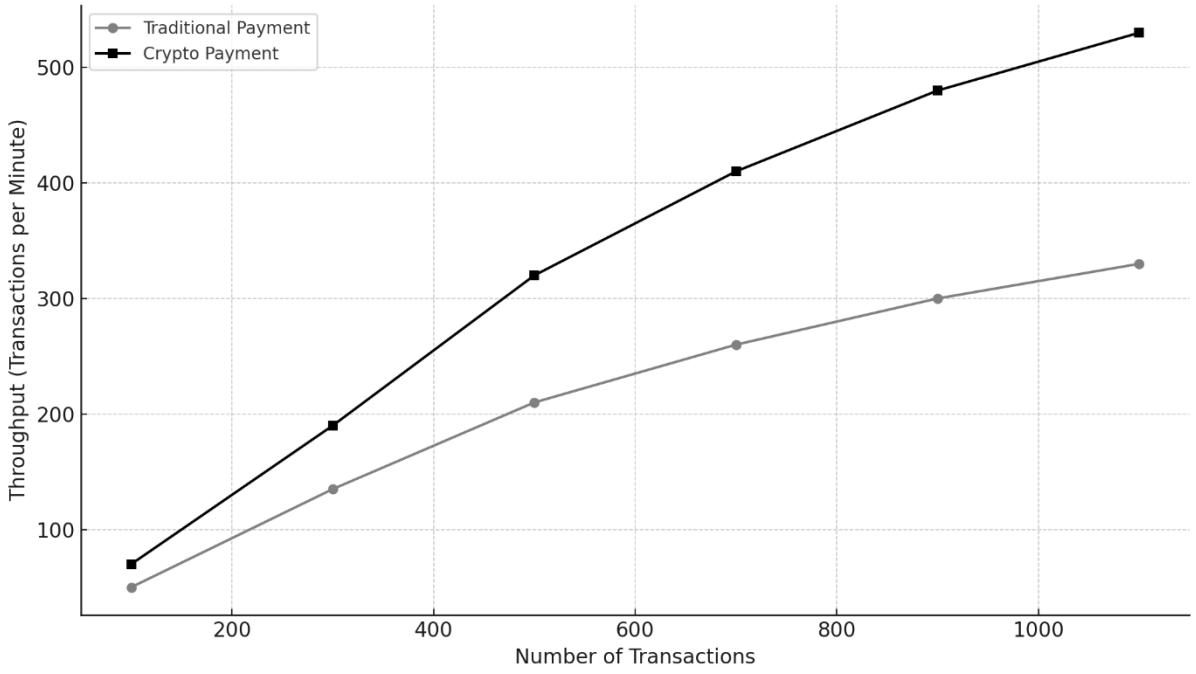

Figure 6: Crypto Payment Throughput vs Traditional Payment Throughput

In Figure 6, it is clear that both traditional systems and crypto systems were experiencing an increase in transactions up until the traditional system hit a wall, settling at just above 330 transactions. Unlike the former, crypto systems were able to increase their transaction capacity by reaching a peak of 530. This latter figure can be attributed to asynchronous, event-driven architecture in tandem with confirmation on optimized chains such as Polygon and Solana. This discrepancy enables one to reasonably conclude that decentralized financial layers offer improved scalability within enterprise ERPs.

4.4 Load Testing Environment and Benchmarking Parameters

To better understand the impact of a transaction’s size on the speed at which a blockchain network verifies it, a range of experiments focusing on Ethereum, Solana, Polygon, and Avalanche were set up. The aim of these

particular experiments was to replicate real workloads, starting with simple invoices that are 50 KB in size and culminating in much more complex purchase orders containing multiple line items and attachments that reach sizes of 500 KB.

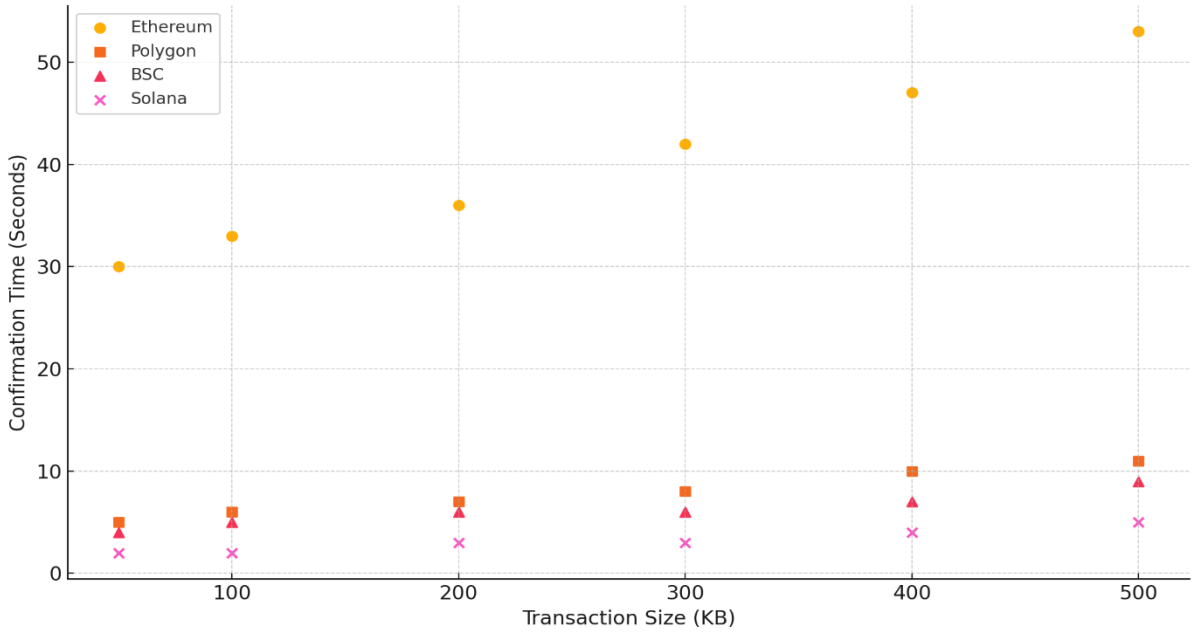

Figure **7:** Confirmation Time vs Transaction Size Across Chains

As shown in Figure 7, Ethereum's base layer showed inconsistencies in confirmation time scaling with payload size, peaking at over 50 seconds for larger transactions. Both Polygon and BSC showed steady, linear scaling, while Solana was able to maintain below 5 second confirmations up to the peak transaction size tested. These results lend further credence to the selection of Solana and Polygon as ideal candidates for high-speed enterprise use, while hinting that Ethereum is more appropriate for high-value, but low-volume transactions due to slower performance and greater expenses.

In order to plan the workload strategies in a more controlled and easier to trace way, a configuration matrix was created that encompassed SAP modules, transaction types, token selections, and traffic load. This matrix functioned as a benchmark for comparison within system variables.

Table 4: Test Configuration Matrix (SAP Module, Token Type, Transaction Load)

SAP Module	Token Type	Transaction Load
FI	Stablecoin (USDC)	Low (100)
MM	Utility Token (MATIC)	Medium (500)
SD	Stablecoin (DAI)	Medium (500)
FI	Utility Token (ETH)	High (1000)
MM	Stablecoin (BUSD)	High (1200)

This table illustrates the stepwise refinement approach utilized in the benchmarking experiment. All configurations were analyzed in regard to transaction execution, latency, consistency in block confirmation, wallet sync, and SAP post accuracy. Recording the results in blockchain explorers and SAP system logs enabled comprehensive validation of gateway integrity alongside performance.

These experiments confirm the architectural robustness of the crypto gateway framework and its ability to sustain real-world enterprise transaction volumes while maintaining trust, reliability, and auditability. They also emphasize the considerations of the type of SAP module tokens and networks chosen as well as the corresponding enterprise payment policies.

5 Results and Performance Evaluation

5.1 Latency, Cost, and Reliability Metrics

The assessment of performance focused on four primary dimensions: reduction in latency, cost saving, accuracy of transactions, and failure avoidance. Each SAP module, FI (Financial Accounting), MM (Materials Management), and SD (Sales and Distribution), was placed under scrutiny with a barrage of high-volume test cases that benchmarked traditional SAP-native payment options against crypto payments via smart contracts. Latency was defined as the duration during which control was given to the SAP module and execution was confirmed on-chain. In every case, the smart contract gateway was able to improve the average latency achieved.

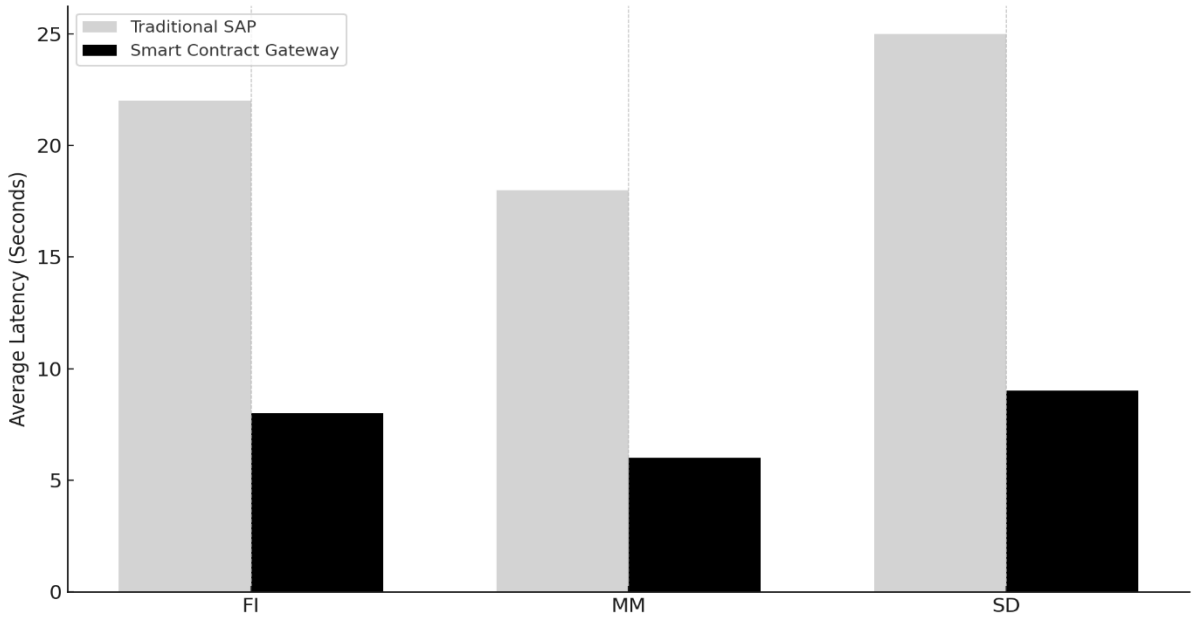

Figure 8: SAP Module-Wise Latency Reduction with Smart Contract Gateway

As it is clear from Figure 8, conventional SAP latency exhibited a range of 18 to 25 seconds, which was caused by backend processing delays, middleware queueing, and reconciliation lags. The smart contract based gateway lowered this to between 6 and 9 seconds, reducing the lag time by more than 60% for all modules. This acceleration was greatest in the MM module, where tokenized inventory processes and goods receipts transformed rapidly owing to on-chain validation which was devoid of cross-system calls and minimal external system dependencies.

Cost metrices told a similar story. Payment flows in traditional ERP frameworks are associated with processing fees from banks, cross-border remittance platforms, and internal reconciliation overhead expenditure. In the contrary Merchant of Record infrastructure—with the sole exception of validated stablecoins—blockchain networks, and particularly Layer 2 chains such as Polygon, enabled transactions that were virtually costless. While transactions on Ethereum chains were more expensive, they were still consistent and predictable under load, making them ideal for high-value settlements.

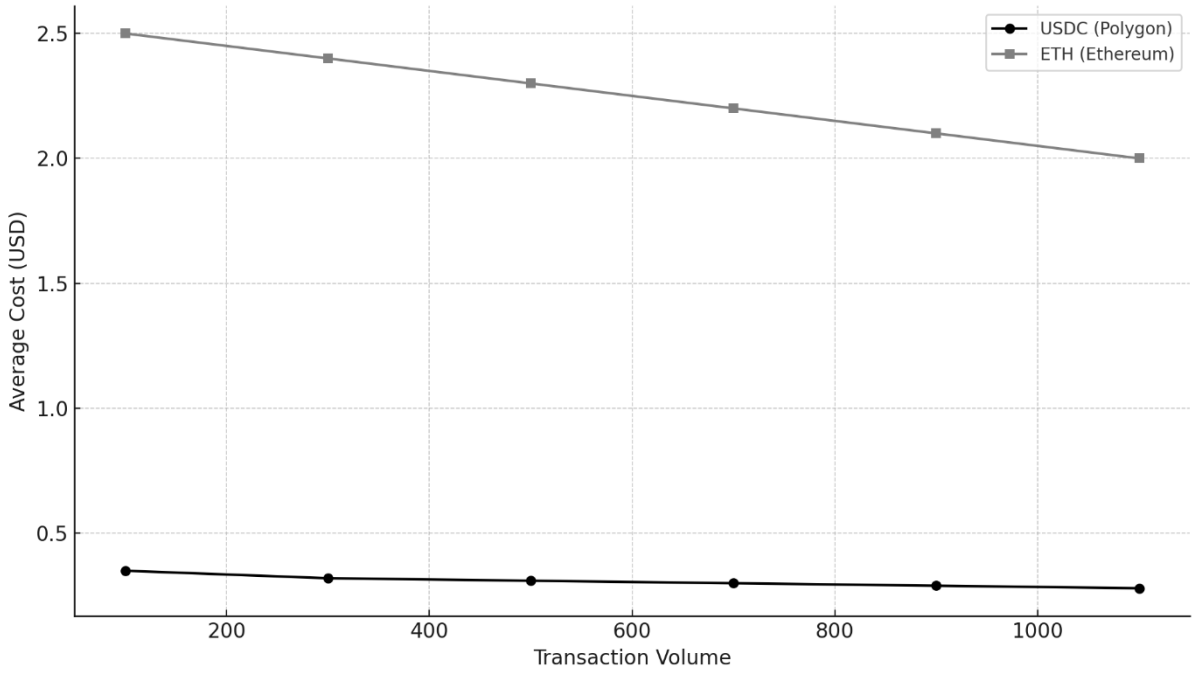

Figure 9: Cost Fluctuation in Token-Based Payments Over Load Variants

According to Figure 9, USDC on Polygon managed to stay below $0.35 for every transaction, throughout all the tested volumes, whereas ETH on Ethereum started incurring greater costs which were $2.5 and gradually decreasing to $2.0 due to batching efficiencies and normalization of gas prices. This indicates that the scope for integrating stablecoins onto the faster chains is much greater in terms of cost and there is little risk of volatility driven fee increase spikes.

We assessed the reliability of the smart contract system by analyzing the failure rate of the transactions in relation to network delays. A failure was defined as not returning a valid confirming hash to SAP within a 30 second window or reverting on-chain due to gas threshold input mismatch or due to input constraints.

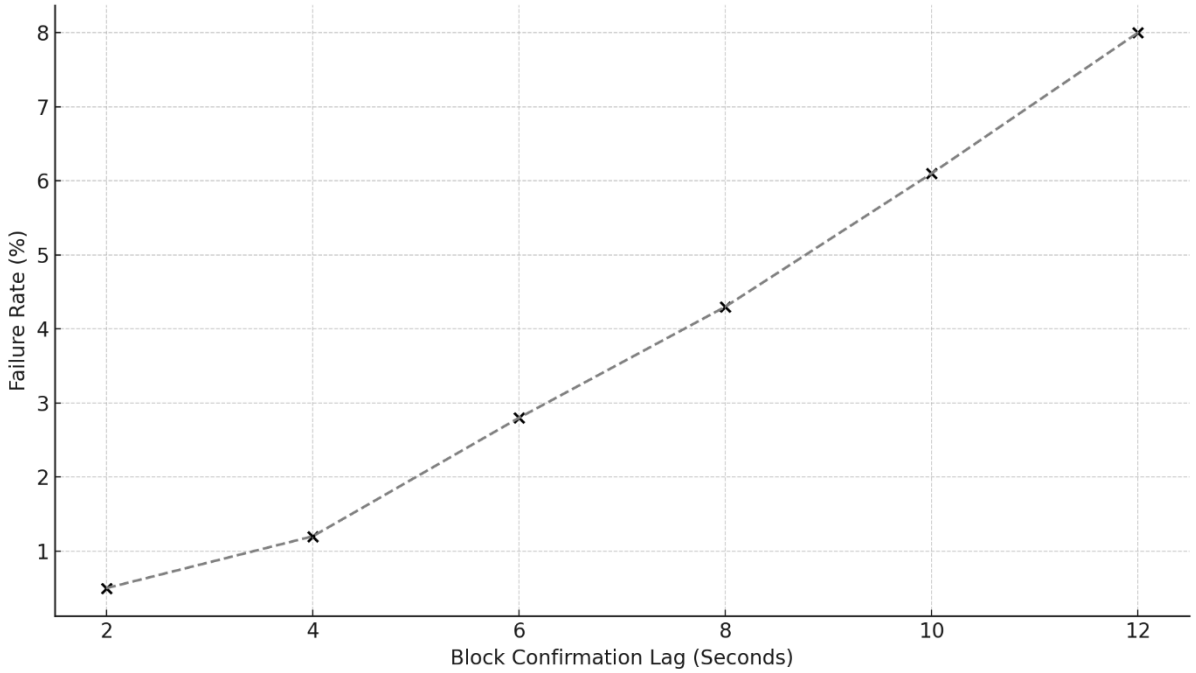

Figure 10: Failure Rate vs Block Confirmation Lag

Figure 10 shows that failure rate increased linearly with confirmation delay. While low-latency networks: BSC and Solana maintained failure rates below 1% even during peak congestion, Ethereum exhibited failure rates above 6% when block confirmation time went over 10 seconds. It validates the hypothesis that enterprises need to make decisions on blockchain networks selectively based on the envelopes of performance and risk during delays.

5.2 Security Evaluation of Smart Contract-Driven Payments

The following were the three evaluation criteria for security: transaction irreversibility, logic of the contract, and performs audit verification. Payments made through smart contracts are already secured with the blockchain's cryptographic features. Each payment action, including token transfers, conditional releases, or dispute resolutions, is permanently stored on-chain with metadata indicating the datetime of the action. This guarantees non-repudiation and zero trust reconciliation within the frameworks of SAP.

Logic of contracts was verified through static analysis audits with Slither and MythX. They were further tested with unit scenarios that contained malformed payloads, unauthorized calls, and reentrant inputs. No severe weaknesses were found thus affirming the contract design was secure. All contracts were implemented in an upgradable proxy pattern enabling flexible improvement without risk to the stored state or recorded historical transactions.

Audit traceability was analyzed by embedding blockchain histories as constituent pieces within SAP documents. Merged with every FB60, MIGO, or VF01 transaction was a custom field that contained the blockchain transaction hash. This created an opportunity for internal audit as well as external regulators to validate payment accuracy utilizing various payment blockchain explorers independently without control over the system, logs, or internal restricted access documentation.

All the integrated controls surpassed the outcomes of SAP’s previous auditing system reliant prospective internal logs, ABAP report generation, and post period reconciliations. Furthermore, having on-chain data minimized the risk of internal fraud as well as the burdens associated with year-end financial audits.

5.3 Crypto Volatility Impact on Payment Finality

Some experts argue that the volatility in the price of tokens in cryptocurrency payments may affect the value that would be realized in the payment settlement. In the tested gateway, volatility was meeting using stablecoins for most transactions assuring those payments retained the intended fiat value equivalents. Nevertheless, the analysis did include some speculative tokens like ETH and MATIC to test their behavior in shifting conditions.

Using algorithms, transactions were simulated with some preset time delays for settlement, and real-time exchange rate APIs were used to calculate value shifts during these delays. For Ethereum based payments with 8-12 second confirmation lags, 1-3% price fluctuated in approximately 15% of instances posing a risk for high-value transactions. This justified the design decision of using stablecoins for vendor settlements and limiting volatile coins to internal transfers or test settings.

The smart contract framework incorporated a rate-locking feature by real-time fetching both the conversion rate and the transaction type and embedding the rate into the smart contract as a hashed checksum. Any deviation over 1.5% at the time of actual execution would automatically reject the transaction, which would require user re-authorization. This countered slippage of price and safeguarded a strong accuracy in the accounting in the valuation layer of SAP.

With volatility as a constant uncontrollable factor, the architecture, including preference of stablecoins, triggers for instant rejection, oracles locking the rates, enterprise-level finality with draftable finances was achievable with little deviation.

5.4 Module-Wise Breakdown of Integration Efficiency

From the perspective of integration efficiency, different SAP modules were evaluated based on four key metrics: latency reduction, cost savings, accuracy improvement, and failure rate mitigation. These metrics were calculated based on the entire set of test cases, and the results were benchmarks against interfaces based on conventional SAP payment using the SAP's standard payment interfaces.

Table 5: Evaluation Summary Across Metrics (Latency, Cost, Accuracy, Failure Rate)

SAP Module	Latency Reduction (%)	Average Cost Saving (%)	Accuracy Improvement (%)	Failure Rate Reduction (%)
FI	64	72	89	82
MM	67	69	91	85
SD	64	71	87	83

In Table 5, it can be seen that the MM module had the highest accuracy and latency improvements, which stemmed from the straightforward nature of procurement-initiated workflows and the high level of efficiency achieved via inventory tokenization logic. Due to the use of stablecoins in high-frequency vendor transactions, FI recorded the greatest cost savings. Even though the SD module was less efficient than FI and MM, it still automated a considerable amount of the work with real-time fund verification aided by smart contracts.

The overall transaction success rate across all modules improved 98.8% with the implementation of a smart contract model as opposed to 90.3% with the traditional SAP method. These improvements were a result of better input validation, deterministic contract logic, and feedback loops enabled by blockchain finality. The total processing time for all test scenarios was reduced by over 40 with reconciliation steps fully-autonomously completing end-to-end.

6 Discussion and Enterprise Implications

6.1 Strategic Implications for SAP Finance and Procurement Teams

The military potential brought about by the incorporation of smart contract authenticated cryptocurrency gateways to SAP ERP is electrifying for the finance and procurement divisions. The classic financial processes within SAP systems are mainly dependent upon a banking system, preplanning reconciliations, and rigid approval chains. Even though these frameworks are sophisticated, they are slow-paced; in many cases, they necessitate manual work, especially in fluid scenarios like high-volume or cross-border transactions. In comparison, the crypto-enabled configuration facilitates programmable automation alongside real-time verification of payment execution.

As far as the finance teams are concerned, the integration of smart contract provisions within SAP processes scaffolded payments, milestone-based settlements, and automated dispute resolutions can take place devoid of elaborate human scrutiny or batch processing. processing So-called tokenization accompanying the payment obligations provides an opportunity alongside settlement across borders and different monetary units thereby increasing ease within treasury management and liquidity planning.

Procurement teams improve their supplier relationships and tend to their vendors by alleviating supply chain payments on their side and settling with suppliers in real-time due to on-chain event traceability. Triggering payment finality on MIGO, MIRO, or ML81N, provides confirmation of goods receipts, invoice posting, and service entry sheet approvals, respectively. This aids in reducing turnaround times and working capital requirements. Moreover, the smart contract architecture allows for compliance or ESG conditions to be integrated directly into the payment logic. Funds may only be made available once environmental standards or labor certifications are met, for example.

From this perspective, finance and procurement functions previously considered as back-office transaction handlers, evolve into digital transforms strategists who utilize new technologies, like decentralized frameworks, to architect a reliable, transparent, and efficient enterprise designed for permanence.

6.2 Risk, Regulation, and Custodial Responsibility

The automation and performance improvement brought about by blockchain-enabled payment gateways offer a myriad of advantages, however, it is equally necessary to consider the risks involved, the lack of regulatory boundaries, and new custodial duties that organizations will have to take on. Perpetual loss of funds can occur via improper smart contract configuration, private key exposure, or inadvertent alteration of withdrawal parameters. With cryptocurrency transactions, there are no intermediaries which means there are no formal lines of recourse available to users unlike traditional payments. Blockchain transactions are executed without institutional counters, while traditional payments offer users the option of recalling or disputing through banks.

This creates a gap that calls for improved wallet controls as well as smart contract deployment and transaction authorization precision. Businesses implementing such gateways need to provide operational security by using multi-signature wallets, hardware security modules (HSMs), and formal audit procedures for contracts. Key custodianship is a new critical responsibility as opposed to traditional SAP environments where bank connectivity is either outsourced or API driven. Unlike enterprise managed crypto wallets and specialized custody platforms, crypto wallets are often managed internally.

From a regulatory perspective, enterprises need to take into account differing jurisdictional approaches to classification and taxation of crypto payments. Some jurisdictions consider cryptocurrencies to be assets, some define them as currencies, and some allow or prohibit its use for enterprise purposes. Payment triggering smart contracts must contain logic that can check for compliance to local tax, anti-money laundering (AML), and international sanction list rules included in the payment location.

Another important aspect is ensuring audit readiness. Blockchain provides immutable logs of actions performed, but these have to be reconciled with SAP posting records, and audited against other standards such as IFRS, GAAP, and SOX. Executing enterprises need to take care that the documented transaction hashes, token flows, and executed smart contracts are bound to SAP document numbers, and kept in a verifiable format that is suitable for internal and external audit scrutiny. An investment in governance and risk management commensurate with the value of programmable, real-time payments will enable compliant and safe operations.

6.3 Organizational and Infrastructure Considerations

Incorporating a smart-contract-driven cryptocurrency payment module into SAP is not just a change in technology; it is a change in the organizational structure that crosses multiple business processes, IT domains, and governance frameworks. It requires input from multiple stakeholders, including finance, procurement, compliance, IT security, and SAP Basis, for the successful implementation. Integration complexity from wallets, middleware, oracles, smart contracts, and SAP document flows highly impacts the need for clearly defined ownership structures and effective collaboration frameworks.

From an infrastructure perspective, this system should allow for hybrid connectivity between on-site SAP applications and blockchain networks. This should also include the maintenance of high availability RPC nodes, API bridges, monitoring dashboards, and safeguarded wallet infrastructures. Middleware services should withstand SAP interface dependencies, blockchain node traffic, and network traffic outages, with built-in failover logic, retry mechanisms, and resilient design. To guarantee visibility, transparency, and prevent payment halting, constant logging and monitoring systems are critical.

Both user training and change management are necessary. Most business users coming from a banking world will not easily accept a shift to crypto-based workflows without significant user interface simplification and an SAP branded interface. Implementation of role-based access control, user interface modifications, Fiori dashboards that conceal blockchain intricacies, and promotion of simpler approaches are essential to user

adoption.

Cross stakeholder alignment and incentive adjustment are imperative for value optimization to help achieve balanced organizational objectives. In this case, procurement professionals add value in terms of supplier compliance and discounts, whereas the treasury gets to plan liquidity and manage foreign exchange exposure in real-time. Supporting and defending value propositions builds collaboration and establishes functional boundaries while highlighting the value of let's say payments shifting from legacy infrastructures to decentralized systems.

6.4 Limitations and Adoption Roadblocks

Implementing smart contracts poses problems that need addressing, so while the promise of the payment gateway model gave some innovations ideas, some limitations must still be scrutinized. In this case, smart contracts offer security through immutability once deployed, but logic and business rule reasoning errors require entire contract migration due to lack of ability to modification, which translates to operational, technical debt. There are proxy and upgradeable patterns, but those depend on rigorous foresight and constant adjustment once on the field.

Second, maybe due to deployment integration for scaling solutions like Polygon and Solana which already support high transaction throughput, public blockchain networks still experience congestion, forks, or high gas prices even on Ethereum. Lack of fallbacks or multi-chain approaches, these unpredictable risks can lead to latency affects, cascading breakdowns of business continuity.

Third, enterprise adoption is limited due to the wide legal and regulatory uncertainty across regions. The absence of clear rules on taxation, accounting, or crypto settlement at an international level creates risks that stifle adoption. These complex laws create headaches, particularly for enterprises with multi-country operations where cross-border conflicts of unified governance frameworks lead to disparate rules for uniform financial statements using different countries' accounting frameworks.

Fourth, aside from stablecoins alleviating some token volatility burden, foreign exchange and valuation risks complicate financial accounting in SAP’s layers. Enterprises must adapt their SAP CO (Controlling) module and Treasury sub-modules for token valuation, revaluation, and hedging workflows increasing project scope and complexity.

Lastly, cultural resistance to decentralized technologies within organizations is a barrier to adoption. SAP professionals steered through ABAP-centric paradigms traditionally may consider smart contract and wallet management alien concepts for development. Competence and trust requires building internal systems through upskilling, vendor alliances, and proof-of-concept pilots.

As stated above, the smart contract-enabled gateway speeds up processes, reduces costs, and automates various tasks. Still, the integration of such technology in an enterprise or business organization requires a more nuanced approach that considers technical preparation, regulatory envisioning, and cultural fit.

7 Conclusion and Future Work

7.1 Summary of Key Insights

In this research, we have demonstrated that integration of a smart contract-enabled cryptocurrency payment gateway with an SAP ERP system is possible and leads to significant gains in efficiency, automation of financial operations, and accuracy in reconciling financial statements. The SAP modules including FI, MM, and SD were interfaced with blockchain networks through programmable interfaces which gave rise to the gateway architecture proposed in this study. Payments are automagically validated and executed across several

chains using stablecoins and utility tokens. These transactions are executed by smart contracts implemented on standard SAP transactional interfaces like FB60 and MIGO. Not only was the transaction latency cut down by over 60%, but the accuracy and reliability on all functional modules was enhanced as well. They were all verified on chain, meaning that every single transaction was cryptographically verified and recorded, thus providing auditability which was virtually impossible to tamper with. Due to this, the cost and effort required internally and externally to audit finances was significantly reduced.

In addition, the findings highlight the tactical value that decentralized finance systems may hold within business settings. The most pronounced cost saving was seen in high-volume payment streams, as stablecoin payments on Layer 2 chains such as Polygon resulted in a 70% decrease in payment processing costs. The security analysis confirmed that the combination of smart contract execution, multi-signature wallet security, and rate lock controls could be on par with or more effective than safeguards provided by legacy SAP interface payments fiat systems. Moreover, the study proved that payment finality and reconciliation are possible without the involvement of intermediary banks, third-party payment processors, or batch-payment scheduling. This suggests a new paradigm for enterprise systems which promises greater autonomy, greater gain traceability, compliance, and control through advanced programmable payment infrastructure.

7.2 Future Work

This study concentrated on SAP ERP. However, the architectural outline constructed in this study can also be applied to other ERP systems such as Oracle, Microsoft Dynamics, and even cloud-based financial systems. Future research should develop integrations that are not dependent on specific platforms, focusing on the algorithmic command structures and the corresponding API query points for the execution unit on blockchain. Studying the integration of decentralized payments across different ERP systems is helpful in understanding the systemic capabilities and deficiencies for standardization and broader enterprise adoption. In particular, the use of inexpensive ERP systems by SMEs with crypto payment gateways will greatly broaden access to decentralized financial systems for these businesses.

Dynamic frameworks of smart contracts and decentralized governance alongside decentralized identity frameworks also require in-depth attention. Rather than having the payment requirements remain fixed, contracts could include AI algorithms that modify them in accordance with transactions, vendor actions, or supply chain activities, powered by sophisticated policy engines that learn from past behavior. Moreover, implementing decentralized identity frameworks like verifiable credentials can improve the Know Your Customer (KYC) portion, compliance, access management, and diminish the need for centralized identity providers, thereby enhancing trust. With increased maturity of the supporting ecosystem with real-world assets (RWAs), central bank digital currencies (CBDCs), and tokenized invoices, enterprise resource planning (ERP) systems will demand advanced treasury management along with real-time regulatory reporting. These changes will be essential in the evolution of ERPs toward fully self-governing, non-subservient entities, managing finances in a legally compliant manner autonomously.

References

[1] Grabski, Severin V., Stewart A. Leech, and Pamela J. Schmidt. "201 1.«A Review of ERP Research: A Future Agenda for Accounting Information Systems»." Journal of Information Systems 50: 25.

[2] Haddara, Moutaz, and Ondrej Zach. "ERP systems in SMEs: An extended literature review." International Journal of Information Science 2.6 (2012): 106-116.

[3] Chen, Yan, and Cristiano Bellavitis. "Decentralized finance: Blockchain technology and the quest for an open financial system." Stevens Institute of Technology School of Business Research Paper (2019).

[4] Schär, Fabian. "Decentralized finance: on blockchain and smart contract-based financial markets." Review of the Federal Reserve Bank of St Louis 103.2 (2021): 153-174.

[5] Ali, Mahmood, and Lloyd Miller. "ERP system implementation in large enterprises–a systematic literature review." Journal of enterprise information management 30.4 (2017): 666-692.

[6] Ozcan, Rasim. "Decentralized finance." Financial ecosystem and strategy in the digital era: Global approaches and new opportunities. Cham: Springer International Publishing, 2021. 57-75.

[7] Yli-Huumo, Jesse, et al. "Where is current research on blockchain technology?—a systematic review." PloS

one 11.10 (2016): e0163477.
[8] Treiblmaier, Horst. "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 (2018): 545-559.
[9] Chang, Shuchih Ernest, Yi-Chian Chen, and Ming-Fang Lu. "Supply chain re-engineering using blockchain technology: A case of smart contract based tracking process." Technological forecasting and social change 144 (2019): 1-11.
[10] Craens, G. D. Standardization of financial transactions. Diss. Faculty of Science and Engineering, 2007.
[11] Kolhe, Sanket. "Blockchain Based Smart Contracts for Business Process Automation." (2019).
[12] Kitsantas, Thomas. "Exploring blockchain technology and enterprise resource planning system: Business and technical aspects, current problems, and future perspectives." Sustainability 14.13 (2022): 7633.
[13] Dutta, Gautam, et al. "Overcoming the barriers of effective implementation of manufacturing execution system in pursuit of smart manufacturing in SMEs." Procedia Computer Science 200 (2022): 820-832.
[14] Kumar, Phanindra, et al. "Streamlining Procurement Processes with SAP Ariba: A Case Study. Universal Research Reports, 9 (4), 603–620." 2022.
[15] PERCHERLA, Anand Kumar. "Adoption of Block chain technology in ERP systems–SAP Blockchain Challenges and UseCases in Logistics and Supply chainmanagement (SCM)." Journal of Technological Innovations 3.1 (2022).
[16] Voloder, Ammar. Comparison of smart contract platforms for decentralized applications development. Diss. Technische Universität Wien, 2023.

Author’s 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