DOYENSYS
VALUE INNOVATION

WHY CUSTOMERS SHOULD RUN
ORACLE EBS ON OCI

Transforming Oracle E-Business Suite with Oracle Cloud Infrastructure and Generative AI

By Narasimharao Karanam
Oracle ACE Associate | Cloud & Data Platform Architect

Introduction

Oracle E-Business Suite (EBS) is one of the most database-intensive enterprise applications in use today. Its performance, stability, and recoverability are tightly coupled with the behavior of the underlying Oracle Database, the efficiency of the I/O subsystem, and the network interconnect between application and database tiers.

When migrating or modernizing an EBS environment, the architecture must deliver **predictable latency, deterministic performance, strong workload isolation, and integrated High Availability (HA) and Disaster Recovery (DR)** capabilities. These are not optional—they are foundational to sustaining enterprise operations.

At the same time, enterprise expectations have evolved. Modern business users are no longer satisfied with static reports and complex navigation. They expect:

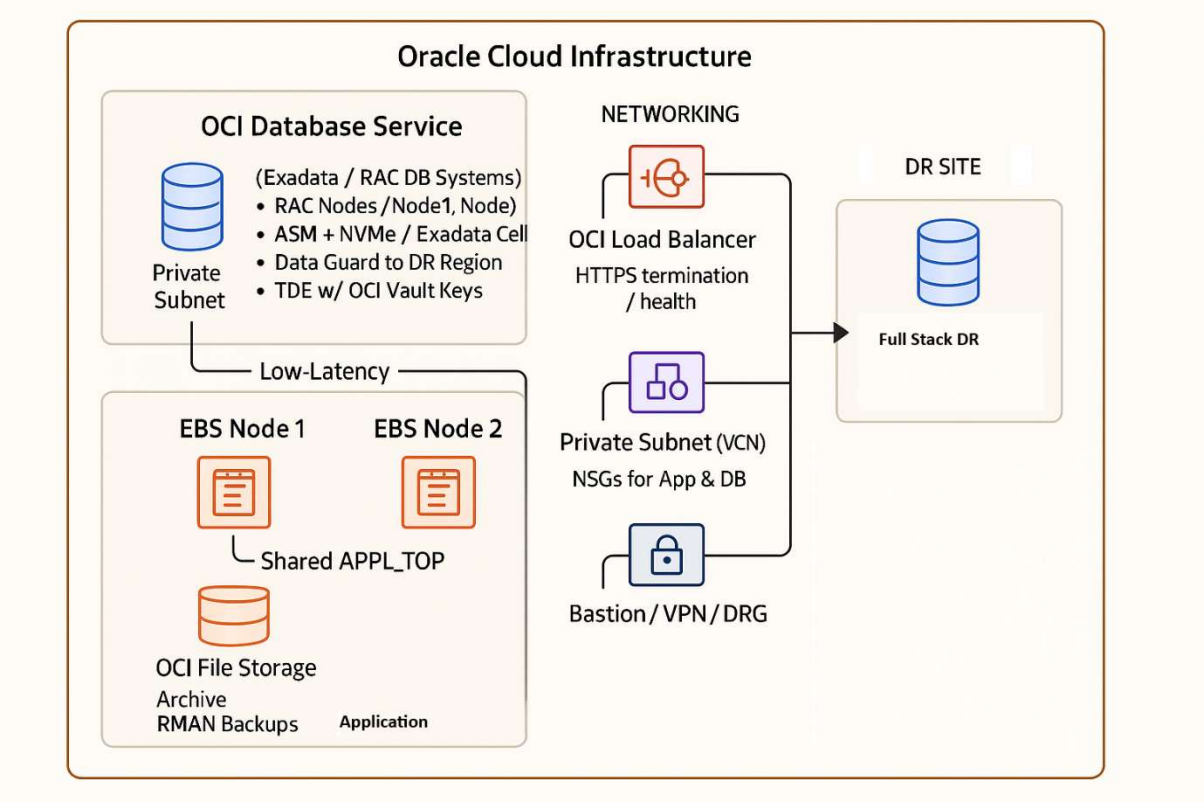
- Secure and scalable infrastructure
- Conversational access to enterprise data
- AI-driven decision support
- Faster, real-time insights instead of delayed reporting

In this e-book, I will share a practical and experience-driven approach to:

1. Why Oracle E-Business Suite is best optimized to run on Oracle Cloud Infrastructure (OCI)
2. How to enable Natural Language Query (NLQ) using OCI Generative AI services
3. How to build AI-powered business solutions using OCI on-demand models

DOYENSYS
VALUE INNOVATION

Why Oracle E-Business Suite is best optimized to run on Oracle Cloud Infrastructure (OCI)



1. Database Tier: Why OCI Database Services Are Engineered for EBS

The EBS database performs heavy OLTP, high-volume concurrent batch processing, and frequent read/write spikes during business events. OCI Database Services (DB Systems and Exadata Cloud Service) provide database-specific hardware and software features that general-purpose virtual machines cannot reproduce.

1.1 Deterministic I/O Throughput and Latency

Exadata Cloud Service

Exadata is the only database platform that applies database-aware intelligence directly in the storage layer. This includes:

- **Smart Scan and Smart Flash Cache** enabling microsecond-level read latency
- **Offload filtering, joins, and projections** reducing CPU work on database nodes
- **Storage Indexes** that skip disk access for nonqualifying data
- **Redo log write optimizations** using persistent memory and flash modules

These accelerate key EBS modules that frequently run large reports or heavy lookup operations, including General Ledger, Subledger Accounting, Order Management, and Costing.

DB Systems

For non-Exadata deployments, DB Systems provide:

- **Local NVMe SSDs** with extremely high write IOPS, ideal for redo/temp
- **Striped OCI Block Volumes under ASM**, scaling linearly with volume count (multi-100K+ IOPS achievable)
- **Dedicated storage performance**, not shared with other compute tenants

This yield stable performance for concurrent programs, payroll processing, and financial close operations.

1.2 Predictable RAC Behavior and Low-Latency Interconnect

EBS customers commonly deploy **Real Application Clusters (RAC)** for node-level HA.

OCI supports RAC on specialized DB shapes with:

- **Dedicated RAC Interconnect** isolated from external traffic
- **High-bandwidth, low-latency interconnect** for Global Cache Service
- **Fast node eviction and rejoin behavior** improving cluster stability

Race conditions and "gc buffer busy" waits reduce significantly when interconnect jitter is eliminated, directly improving user session response times.

1.3 Integrated Physical Standby and Failover Automation

Using **Data Guard ,Active Data Guard , Full Stack DR**:

- Cross-AD replication protects against Availability Domain outages
- Active Data Guard allows real-time read-only queries offloaded from primary
- Fast-Start Failover (FSFO) reduces RTO without human intervention

Unlike a VM-based do-it-yourself database, OCI DB Services include tooling, automation, validation, and consistent patching pipelines.

1.4 Autonomous Backup, Patching, and TDE Key Management

The most common production incidents in on-prem EBS DBs come from:

- Incorrect patch application

- Failed or untested backups
- Misconfigured TDE key management

OCI Database Services mitigate these with:

- Automated patching with pre-checks
- Incremental-forever RMAN backups integrated with Object Storage
- Mandatory encryption with TDE keys stored in **OCI Vault (HSM-backed)**

This ensures operational consistency and audit compliance without manual intervention.

2. Application Tier: Why OCI Compute VM Shapes are Optimal for EBS Web and Forms

EBS application tiers (oacore, forms, admin, concurrent managers) require predictable CPU cycles, stable network throughput, and a shared APPL_TOP. OCI Compute delivers these characteristics without overhead.

2.1 CPU Predictability for JVM and Forms Runtime

EBS uses:

- **OACore JVMs** for web rendering and concurrent program invocation
- **Forms runtime processes** requiring consistent per-core performance

OCI Compute ensures:

- **Non-oversubscribed OCPU allocation**
- Stable clock speeds suitable for latency-sensitive JVM workloads

You can size compute nodes vertically (bigger shapes) or horizontally (more nodes) depending on concurrency profiles.

2.2 Low-Latency Network Fabric Between App & DB

When application and database tiers reside in the same VCN and AD:

- JDBC round trips remain <1 ms
- Forms message exchanges avoid packet loss
- EBS sessions become more responsive

This setup outperforms hybrid or cross-cloud architectures where latency variability degrades user experience.

2.3 Shared APPL_TOP with OCI File Storage (FSS)

EBS requires:

- Shared filesystem for APPL_TOP
- Concurrent manager log/archive output
- Patch staging, cloning, and code-level synchronization

OCI FSS provides:

- NFS with high throughput (hundreds of MB/s)
- Elastic performance modes scaling based on workload
- Zero management overhead (no NAS clusters, no failover scripting)

It integrates cleanly with multi-node EBS application tiers.

2.4 Built-In Load Balancing and Autoscaling

OCI Load Balancer provides:

- Layer 4/7 health checks for oacore/forms
- Connection draining for patch cycles
- Auto-scaling of the app tier based on OCPU or heap usage metrics

This ensures stable performance during seasonal or month-end peaks.

3. Security: Layered Controls Built for Enterprise Applications

EBS implementations must meet strict audit and compliance requirements. OCI natively provides multi-layered security controls that map directly to EBS architectural boundaries.

3.1 No Public Exposure for DB or App Tiers

By using only:

- Private subnets
- Network Security Groups (NSGs)
- Service gateway / Bastion for controlled access

you ensure zero inbound internet exposure for both app and DB tiers.

3.2 Vault-Managed Encryption Keys and TDE Separation

Database encryption leverages:

- TDE (Transparent Data Encryption)

- Master keys stored in **OCI Vault**
- HSM-level FIPS 140-2 compliance

This satisfies Segregation of Duties (SoD) and key custody requirements.

3.3 Identity Governance with OCI IAM

Different administrators can be fully isolated using IAM policies:

- DBA
- App Admin
- Network Admin
- Security Admin

Unlike on-prem systems, these boundaries are enforced by cloud infrastructure, not operating system conventions.

4. High Availability Architecture for EBS on OCI

A production-grade deployment includes HA at three layers:

4.1 Database HA

- RAC on Exadata or VM DB Systems
- Cross-AD/Region Data Guard
- Redo transport integrity checks

4.2 Application HA

- Multiple oacore/forms/admin nodes
- Shared APPL_TOP on FSS
- LBaaS distributing sessions

4.3 Infrastructure HA

- Multiple Availability Domains
- Fault Domains for anti-affinity
- DR replicating to second region via Data Guard

This structure creates resilience from component-level failure up to regional disasters.

5. Consolidated Technical Justification

Hosting EBS Databases on OCI DB Services delivers:

- Deterministic I/O performance via Exadata or NVMe/block volume ASM
- RAC stability and predictable interconnect latency
- Native Data Guard DR with automated failover
- Automated backup/patching with TDE enforced
- Dedicated storage → no performance disruption under load

Hosting EBS Apps on OCI Compute delivers:

- Guaranteed CPU cycles for JVM/Forms
- <1 ms latency to DB tier inside same VCN/AD
- High-throughput NFS via FSS for APPL_TOP
- Native load balancing, autoscaling, and logging
- Zero public exposure with NSG-based isolation

Together, this architecture provides the highest degree of reliability, performance consistency, and operational safety for Oracle E-Business Suite.

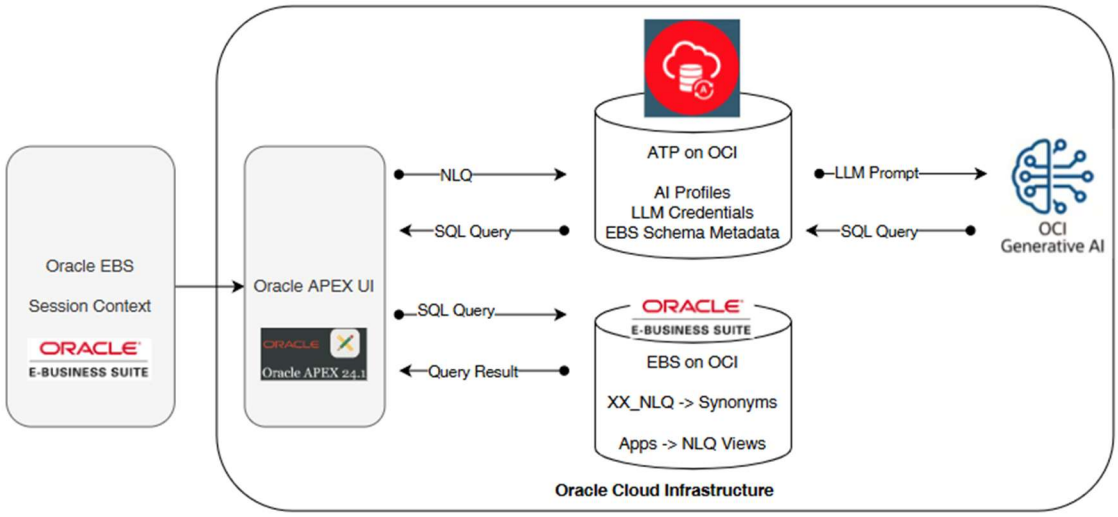
While OCI provides the optimal foundation for running Oracle EBS with performance, security, and resilience, the next evolution is not just infrastructure modernization—but how users interact with the system.

This is where Generative AI introduces a transformational shift.

DOYENSYS
VALUE INNOVATION

How to enable Natural Language Query (NLQ) using OCI Generative AI services

"Ask EBS" typically refers to the Natural Language Query (NLQ) functionality recently introduced to Oracle E-Business Suite (EBS) R12.2, which allows users to interact with EBS data using AI in plain English, and get reports output.



Below is the step-by-step technical guide for setting up Ask EBS NLQ to enable GenAI reporting, based on Oracle recommendations (MOS Note 3059877.1).

Technical Implementation Steps for Ask EBS NLQ:

1. Prerequisites Check

- **EBS Version:** Oracle E-Business Suite Release 12.2 or higher.
- **Database:** Oracle Database 19c or higher.
- **Required Patches:** Apply AD and TXK Delta 16 or higher.
- **APEX Integration:** Oracle APEX (Application Express) must be installed and configured in your EBS environment.
- **AI Access:** A subscription to Oracle Cloud Infrastructure (OCI) Generative AI services (e.g., Llama 3.1).

2. Configure Autonomous Database (ADB)

- **Link Databases:** Set up a database link from your EBS 12.2 database to an Oracle Autonomous Database (23ai or 19c).
- **Enable Select AI:** Configure the Autonomous Database to use Select AI, which enables SQL generation from natural language.

3. Setup Oracle APEX Workspace

- **Create Workspace:** Log in to APEX Administration (/ords/apex_admin) and create a new workspace (e.g., EBS_APEX).
- **Map Schema:** Map the workspace to the APPS schema (or a custom schema with APPS access).

4. Install Ask EBS Application

- **Deploy App:** Import the Ask EBS application (provided by Oracle or through partner solutions) into your new APEX workspace.
- **Configure Web Services:** Configure REST services within APEX to connect to the OCI Generative AI model.

5. Configure Security (VPD)

- **Implement Security:** Enable Virtual Private Database (VPD) policies to ensure that the NLQ respects existing EBS role-based security.
- **Data Access:** Ensure the user asking the questions can only see data they are authorized to access within the standard EBS forms.

6. Register/Map Metadata Views

- **Create Views:** Create Oracle SQL views for the business domains you want to query (e.g., APPS.XX_ORDER_MANAGEMENT_V).
- **Update Prompting:** Register these views within the Ask EBS tool to help the LLM understand table structures and columns.

7. Test and Validate

- **Test Queries:** Ask natural language questions like "Show me all the outstanding orders for customer XYZ".
- **Verify SQL:** Check the generated SQL in the APEX console for accuracy, as per security compliance.

Key Components of Ask EBS

- **Select AI:** Translates English to SQL.
- **OCI GenAI (LLM):** Processes the question.
- **APEX:** Provides the user interface.

How to Build AI-Powered Business Solutions using OCI On-Demand Models

From Reports to Intelligence

In a typical Oracle E-Business Suite (EBS) environment, business users rely heavily on reports, dashboards, and manual analysis to understand operational issues.

For example:

- An AP user sees an invoice on hold—but does not know *why*
- A finance user sees a variance—but must manually trace journals
- A treasury user sees balances—but cannot predict cash risk

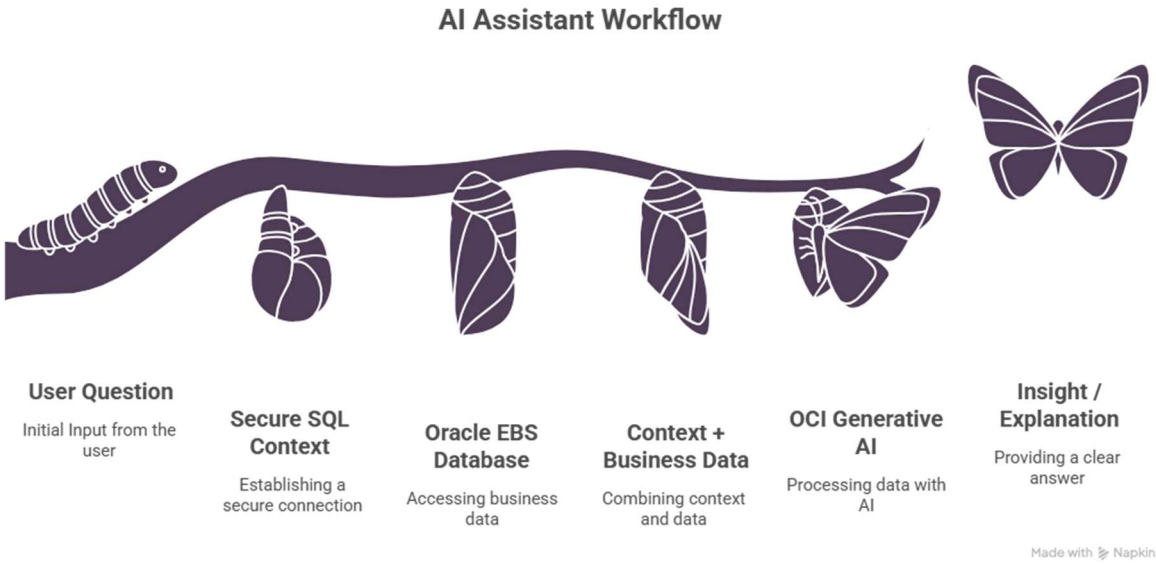
This leads to:

- Delayed decision-making
- Dependency on technical teams
- Time-consuming investigations

Generative AI fundamentally changes this paradigm by enabling systems to not only present data, but also explain what is happening and recommend next actions.

This is where OCI on-demand models bring real value.

Key Design Principle (Foundation of the Architecture)



One of the most critical design decisions in building AI solutions on EBS is:

❗ **Generative AI should never directly query EBS transactional tables**

Instead, we introduce a **controlled data layer**.

Why this matters:

- Protects sensitive data
- Ensures consistent and accurate responses
- Prevents performance impact on EBS
- Enables governance and auditing

AI consumes curated datasets, AI does not access raw EBS schema

This principle is the backbone of a **secure and scalable AI architecture**

Architecture Overview (Explained Simply)

At a high level, the solution consists of four layers:

1. Oracle EBS Database

The source of truth containing transactional data.

2. AI Dataset Layer (SQL Views)

Custom views created to:

- Simplify complex EBS tables
- Provide business-friendly structure
- Control data exposure

3. Middleware Layer (Python / API)

Acts as a bridge:

- Fetches data from EBS
- Builds AI prompts
- Calls OCI Generative AI

4. OCI Generative AI (On-Demand Models)

Processes the request and generates:

- Explanation
- Insights
- Recommendations

Step-by-Step Implementation

Step 1: Define the Business Use Case

Start with a real business problem—not technology.

Examples:

- Why is an invoice on hold?
- Will we have enough cash next month?
- Why did this account balance change?

This ensures the solution delivers **business value**, not just technical output.

Step 2: Create the AI Dataset Layer

This is the most important step.

Instead of exposing raw tables, create **curated SQL views**.

Example: AP Invoice Dataset

```
CREATE OR REPLACE VIEW XX_AP_INVOICE_AI_V AS
SELECT
    aia.invoice_id,
    aia.invoice_num,
    aps.vendor_name,
    aia.invoice_amount,
    aia.invoice_date,
    NVL(aph.hold_lookup_code, 'N') hold_flag,
    aph.reason hold_reason,
    aia.payment_status_flag
FROM ap_invoices_all aia
LEFT JOIN ap_suppliers aps ON aia.vendor_id = aps.vendor_id
LEFT JOIN ap_holds_all aph ON aia.invoice_id = aph.invoice_id;
```

Why this is important:

- Hides unnecessary technical complexity
- Provides clean input to AI
- Improves response quality

Step 3: Build the Middleware Layer

This layer connects EBS to OCI GenAI.

Responsibilities:

- Query the dataset
- Prepare structured input
- Call AI model

Example:

```
import oracledb
import oci

def get_invoice_data(invoice_num):
    conn = oracledb.connect(user="apps", password="pwd",
dsn="host:1521/ORCL")
    cursor = conn.cursor()

    cursor.execute("""
        SELECT * FROM XX_AP_INVOICE_AI_V
        WHERE invoice_num = :1
    """, [invoice_num])

    return cursor.fetchall()
```

Step 4: Construct Business-Aware Prompts

This is where raw data becomes intelligence.

Example Prompt:

You are a finance expert.

Analyze the following invoice:
Invoice Number: 87214
Amount: 25000
Hold Reason: 3-Way Match Failure

Provide:
1. Root cause
2. Business impact
3. Recommended resolution
4. Risk if unresolved

Why prompt design matters:

- Guides AI behavior
- Ensures consistent outputs
- Aligns with business expectations

Step 5: Invoke OCI On-Demand Model

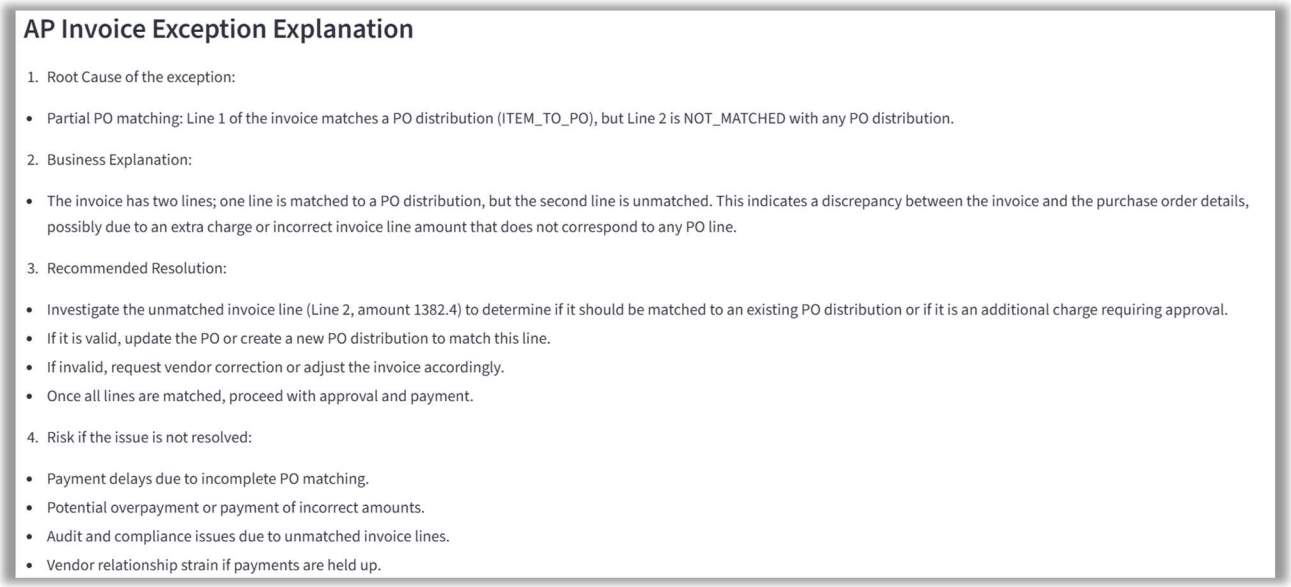
OCI provides multiple models (Cohere, Llama, Gemini).

Example:

```
response = genai.generate_text(
    generate_text_details={
        "model_id": "cohere.command-r-plus",
        "prompt": prompt,
        "max_tokens": 500
    }
)
```

Step 6: Present Output to Users

The output should not be raw text—it should be actionable insight.



AP Invoice Exception Explanation

1. Root Cause of the exception:

- Partial PO matching: Line 1 of the invoice matches a PO distribution (ITEM_TO_PO), but Line 2 is NOT_MATCHED with any PO distribution.

2. Business Explanation:

- The invoice has two lines; one line is matched to a PO distribution, but the second line is unmatched. This indicates a discrepancy between the invoice and the purchase order details, possibly due to an extra charge or incorrect invoice line amount that does not correspond to any PO line.

3. Recommended Resolution:

- Investigate the unmatched invoice line (Line 2, amount 1382.4) to determine if it should be matched to an existing PO distribution or if it is an additional charge requiring approval.
- If it is valid, update the PO or create a new PO distribution to match this line.
- If invalid, request vendor correction or adjust the invoice accordingly.
- Once all lines are matched, proceed with approval and payment.

4. Risk if the issue is not resolved:

- Payment delays due to incomplete PO matching.
- Potential overpayment or payment of incorrect amounts.
- Audit and compliance issues due to unmatched invoice lines.
- Vendor relationship strain if payments are held up.

Example Output

Real Business Use Cases

These are not theoretical—they are practical and widely applicable

AP Invoice Exception Explainer

- Explains invoice holds
- Identifies root cause
- Suggests resolution

Cash Forecast AI

- Predicts liquidity position
- Identifies upcoming risks
- Provides treasury recommendations

GL Variance Explainer

- Explains account changes
- Identifies key drivers
- Highlights anomalies

Order Status Narrator

- Detects delays
- Explains operational issues
- Recommends corrective actions

Model Selection Strategy

Choosing the right model is important:

Scenario	Recommended Model
Finance insights	Cohere Command R+
Cost-sensitive use cases	Llama
Advanced reasoning	Gemini

Security and Governance (Non-Negotiable)

To ensure enterprise readiness:

- Use **read-only SQL views**
- Apply **VPD for data access control**
- Avoid sending sensitive data to LLM
- Log all AI interactions
- Restrict prompts to defined use cases

This is not about adding AI to EBS. It is about transforming how users interact with ERP systems.

Conclusion

Oracle E-Business Suite has been the backbone of enterprise operations for decades - powering finance, supply chain, and critical business processes with reliability and scale.

However, the expectations from enterprise systems have fundamentally changed.

Organizations today are not just looking for systems that can **process transactions**, but platforms that can:

- Deliver real-time insights
- Explain business outcomes
- Enable faster and smarter decisions

Oracle Cloud Infrastructure provides the **ideal foundation** for this transformation by aligning natively with the architecture of Oracle EBS - delivering predictable performance, integrated high availability, and enterprise-grade security.

Building on this foundation, OCI Generative AI introduces a new interaction model:

- From navigating screens → to asking questions
- From analysing reports → to receiving explanations
- From reactive operations → to proactive decision-making

By combining:

- OCI's infrastructure capabilities
- Secure, controlled data architectures
- On-demand Generative AI models

Organizations can transform Oracle EBS into an **intelligent, conversational, and insight-driven enterprise platform - without disrupting existing systems**.

About the Author

Narasimha Rao Karanam is a seasoned Oracle Cloud and Applications Architect with over two decades of hands-on experience in Oracle E-Business Suite, Oracle Cloud Infrastructure (OCI), and enterprise cloud transformations. As a thought leader and practitioner, he has led complex migrations and modernization projects across industries, including manufacturing, healthcare, and public sector.

Currently serving as **Service Delivery Manager - Cloud Services at Doyensys**, Narasimha helps enterprises achieve digital transformation through strategic cloud adoption, automation, and performance optimization. His expertise spans Oracle ERP, Cloud DevOps, architecture design, and IT leadership. He holds certifications as an **Oracle Certified Cloud Infrastructure Architect**, **Cloud Database Architect**, and **AI Architect**.

He is passionate about sharing practical knowledge and simplifying cloud journeys. This e-book reflects his real-world insights and commitment to empowering teams with tested, tool-driven strategies.

Connect with him on LinkedIn: linkedin.com/in/narasimha-rao-karanam-513b5b5

Connect with Us

Are you planning your Oracle E-Business Suite and database modernization or migration to Oracle Cloud Infrastructure?

Doyensys has successfully supported organizations across industries in transforming their mission-critical workloads - delivering reduced downtime, improved availability, and scalable, automation-driven cloud operations.

With deep expertise in Oracle EBS, database platforms, and OCI, our team brings a practical, experience-led approach to every engagement - helping you accelerate your cloud journey with confidence.

If you would like to explore how this approach can be applied to your environment, feel free to connect with us:

📧 Email: info@doyensys.com
🌐 Website: www.doyensys.com
🔗 LinkedIn: Doyensys

Make your cloud transformation simple, efficient, and successful - with Doyensys.