Generative & Agentic AI Reference Architecture

Surendra Panpaliya

Agenda

AI Reference Architecture

Pharma AI Architecture Overview

Multi-layered Structure

Progressive Phases Approach

AI Reference Architecture

Blueprint or high-level design framework

Defines how different AI components work together

From data sources and

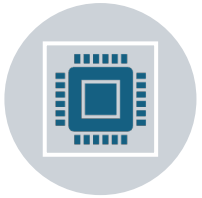
model pipelines to APIs,

governance, and deployment.

AI Reference Architecture

Architectural Skeleton

for AI systems that ensures



scalability, compliance, and

integration across

business and technical layers.

AI Reference Architecture

A reusable, technology-agnostic blueprint

that outlines **components, data flows,**

governance, and interactions

required to build, deploy, and operate

AI solutions responsibly and at scale.

Typical Layers in an AI Reference Architecture

Layer	Key Components	Example in Healthcare/Pharma
Data Layer	Data ingestion, integration, lakehouse, metadata	EHR, LIMS, CDMS, IoT devices
Model/ML Layer	Model training, evaluation, registry, versioning	Disease prediction models, RAG pipelines
AI Services Layer	LLMs, RAG, Agents, NLP, Computer Vision	Medical document summarization, PV case triage

Typical Layers in an AI Reference Architecture

Layer	Key Components	Example in Healthcare/Pharma
Application Layer	APIs, dashboards, chatbots, workflows	Copilot in Excel, Power BI dashboards
Governance Layer	Security, audit, explainability, compliance	HIPAA, GxP, 21 CFR Part 11, ALCOA+
Ops Layer	MLOps, LangOps, monitoring, feedback	Continuous validation, retraining, alerts

Generic AI Reference Architecture Layers

Comprises several layers

Governance and Security integrated

horizontally across all layers

to ensure compliance, trust, and resilience

throughout the entire AI lifecycle.

Generic AI Reference Architecture Layers

Comprises several layers

Governance and Security integrated

horizontally across all layers

to ensure compliance, trust, and resilience

throughout the entire AI lifecycle.

Problem Statement

Bank ABC is looking to set up an

AI CoE and wants to develop a Technology roadmap.

First ask is to develop a reference architecture

for Gen AI including all major components

like LLMs , RAG , Vector DBs , API , deployment etc.

Problem Statement

Including consideration on both

on prem for sensitive data and

cloud for public data.

Assumptions

As the ABC Bank is a fictional entity,

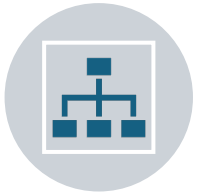
it would be important to have some assumptions

to simulate an actual organisation allowing

for a more complete reference architecture.

Assumptions

Bank ABC has an extensive data solution



following strict governance guidelines.

Sensitive private data such as

transactions and personally identifiable information

is stored securely on-premise and

is not moved to the cloud.

Assumptions

BANK ABC HAS A DEDICATED TEAM OF

DATA SCIENTISTS AND AI RESEARCHERS

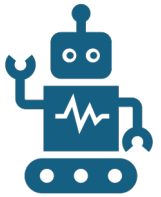
WHO ARE EXPERIENCED IN AI.

Assumptions

The bank has a robust

IT infrastructure that

can support the deployment



of AI models.

Assumptions

Bank ABC is ready to adopt

a hybrid cloud model, with sensitive data

remaining on-premise and

other data potentially

moving to the cloud.

Assumptions

The bank is committed

to adopting AI and is willing

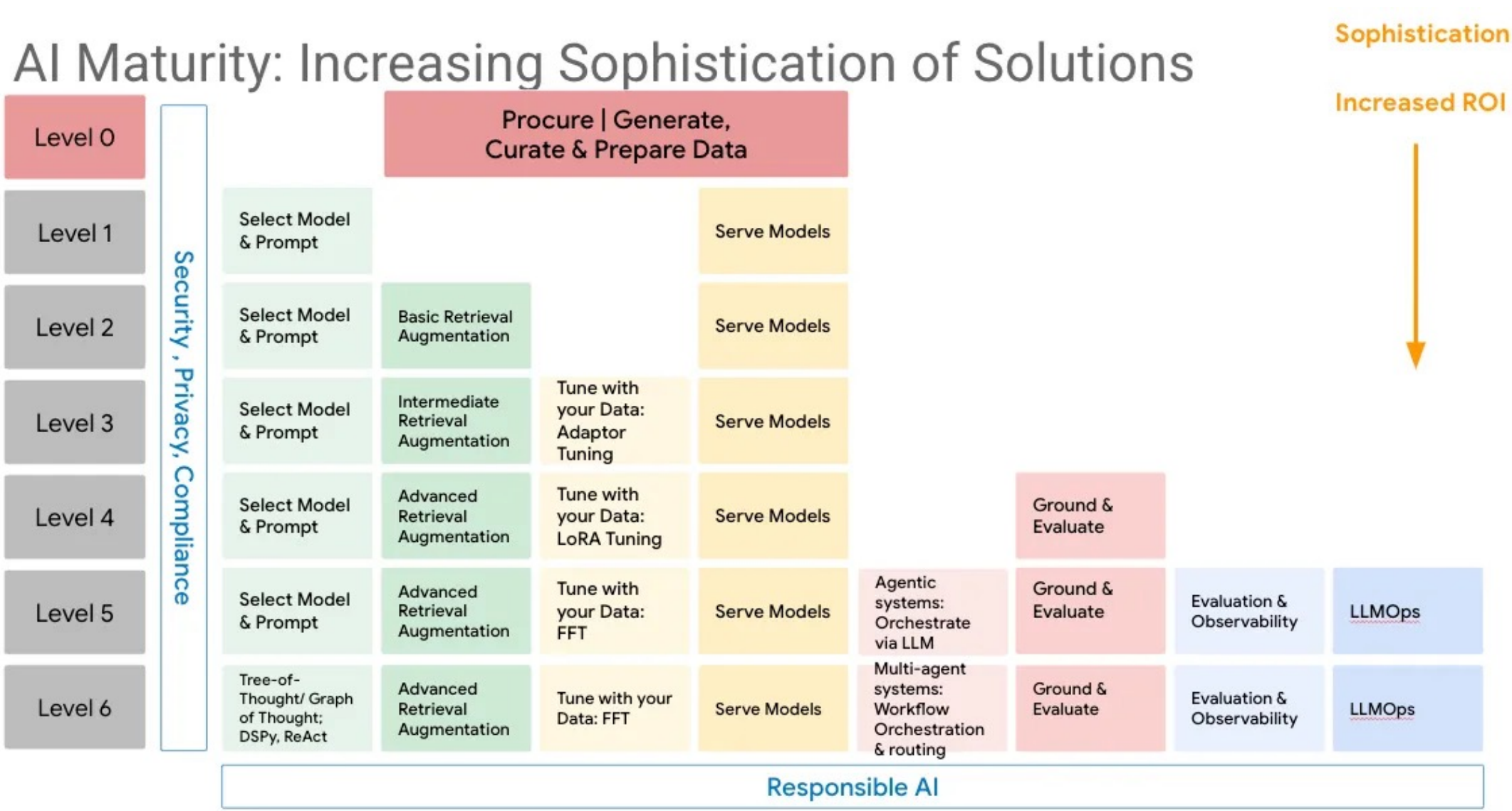
to invest in the necessary resources.

Organisational Maturity

There are 7 levels for

organisational maturity.

AI Maturity: Increasing Sophistication of Solutions



Level 0

If the primary goal or capability is

to collect and organise data

for future GenAI initiatives,

the organisation is likely at

Level 0.

Level 0

Data of course is the foundational element

that fuels AI;

whether predictive AI or generative AI.

Level 1 & 2

If the focus is on using GenAI

for basic tasks like

content generation ,

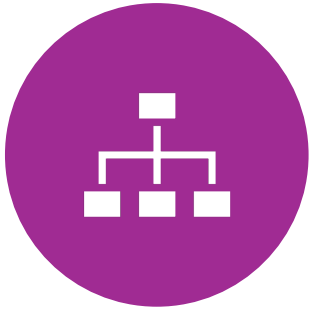
summarising content.

Level 1 & 2

QUESTION ANSWERING USING THE BASE CAPABILITY

KNOWLEDGE OF THE FOUNDATION MODEL

BEING SERVED OR TO INFORMATION RETRIEVAL,



THE ORGANISATION MIGHT BE AT LEVELS 1 OR 2.

Level 3 & 4

Organisations looking to

customise GenAI models

with their data or ensure the quality and

relevance of outputs are

likely at Levels 3 or 4.

Level 5 & 6: Multi-agent systems, advanced reasoning.

For complex use cases requiring

multi-agent systems, advanced reasoning,

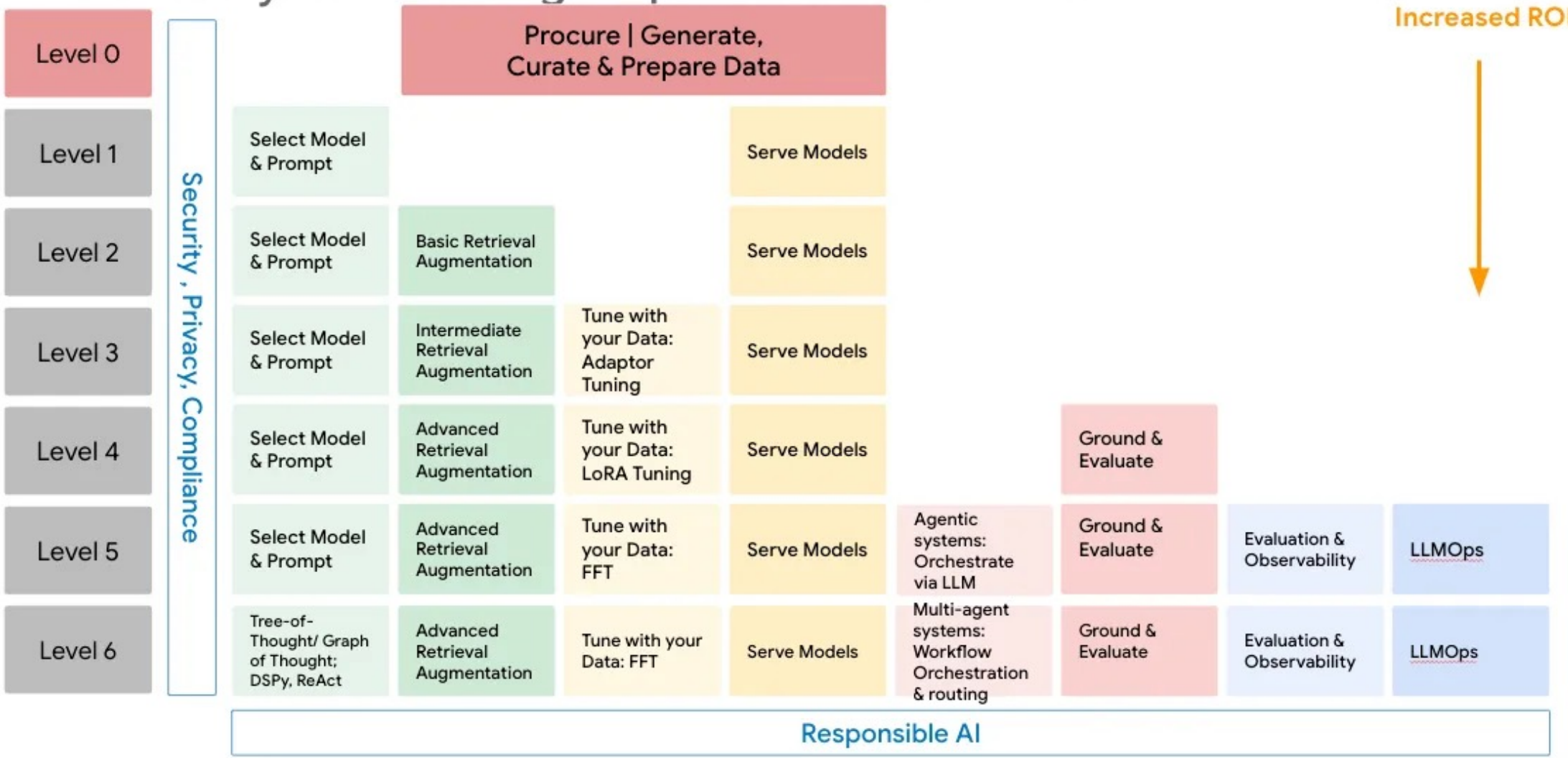
or responsible AI practices,

organisations might be aiming for Levels 5 or 6.

AI Maturity: Increasing Sophistication of Solutions

Sophistication

Increased ROI



Maturity Phases

Phase 1 — Foundation Building

Phase 2 — Intermediate Implementation

Phase 3 — Advanced Deployment

Phase 1 — Foundation Building

Level 0

Primary focus

Setting up the basic infrastructure

Integration layer,

Agent layer

Existing data layer

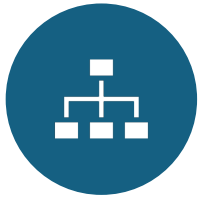
Phase 1 — Foundation Building

Involves collecting and organizing data

for future GenAI initiatives.

Data, being the foundational element

that fuels AI, needs to be



well-structured and easily accessible.

Phase 2 — Intermediate Implementation

For organisations at Levels 1, 2, 3, and 4,

the focus should shift to

more advanced components.

This includes setting up

the RAG layer and the Model layer.

Phase 2 — Intermediate Implementation

During this phase, the organisation

would start using GenAI for basic tasks

like content generation, summarising content,

question answering using the base capability and

knowledge of the foundation model

being served or to information retrieval.

Phase 2 — Intermediate Implementation

To customise GenAI models

with their data or ensure the quality and

relevance of outputs should

also be considering this phase.

Phase 3 — Advanced Deployment

This phase is for organisations at

Levels 5 and 6, aiming for complex

use cases requiring multi-agent systems,

advanced reasoning, or

responsible AI practices.

Phase 3 — Advanced Deployment

Involves setting up the

MLOps layer,

Tuning Layer, and

enhancing

Observability layer.

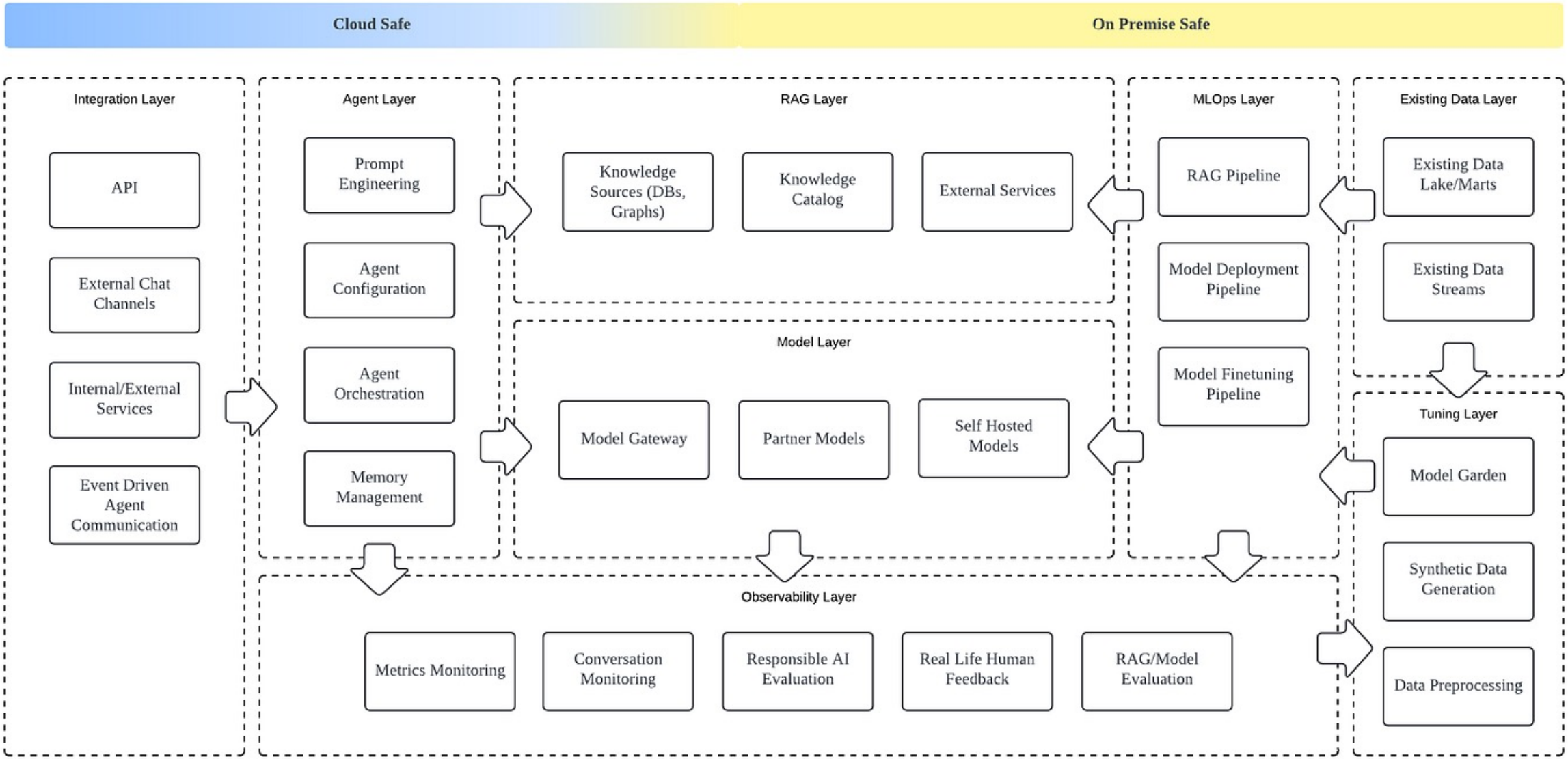
Phase 3 — Advanced Deployment

The organisation should have

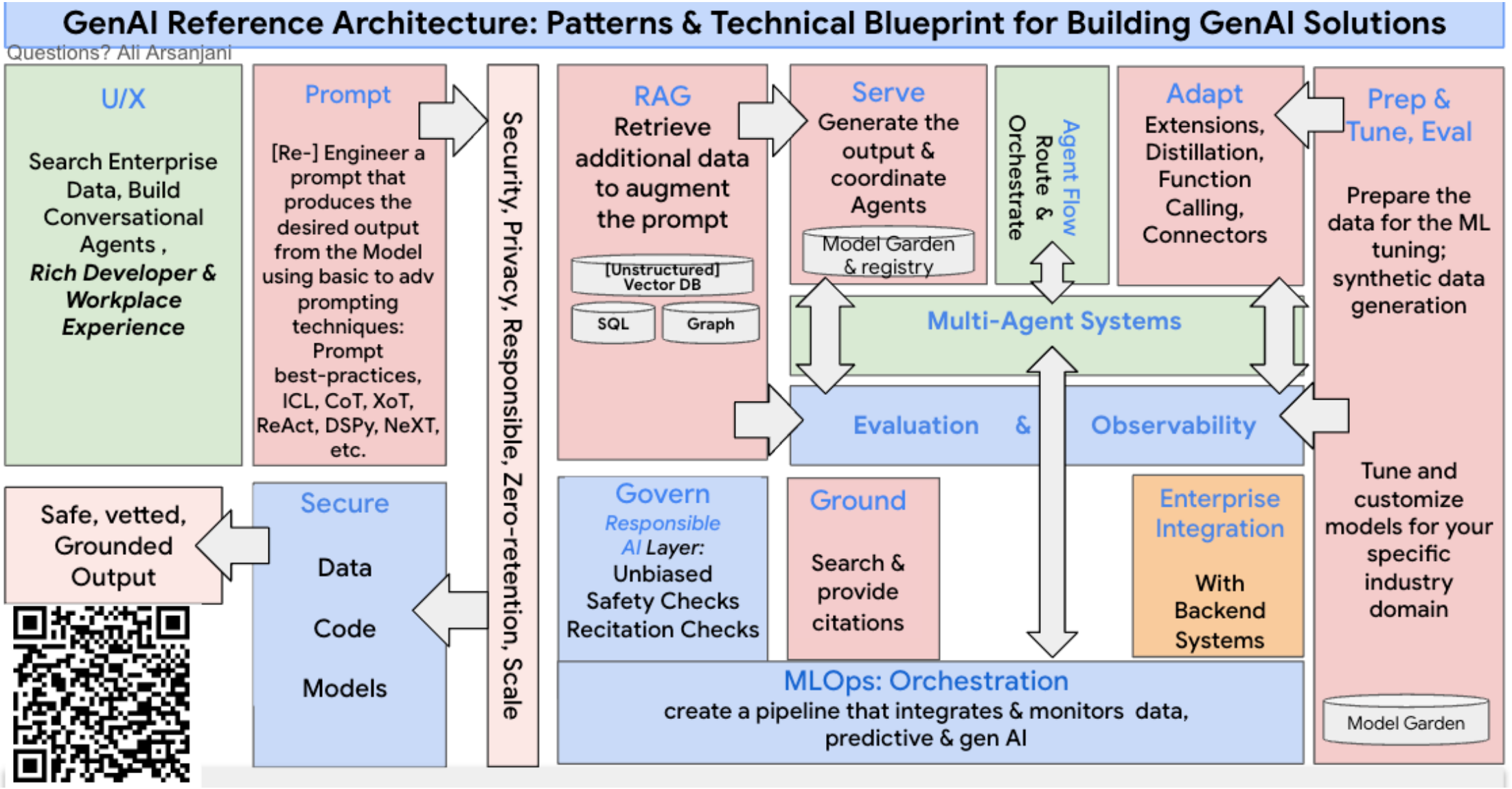
a complete, functioning GenAI system

that’s capable of

handling complex tasks.


Cloud Safe
On Premise Safe
Integration Layer
API
External Chat Channels
Internal/External Services
Event Driven Agent Communication
Agent Layer
Prompt Engineering
Agent Configuration
Agent Orchestration
Memory Management
RAG Layer
Knowledge Sources (DBs, Graphs)
Knowledge Catalog
External Services
Model Layer
Model Gateway
Partner Models
Self Hosted Models
MLOps Layer
RAG Pipeline
Model Deployment Pipeline
Model Finetuning Pipeline
Existing Data Layer
Existing Data Lake/Marts
Existing Data Streams
Tuning Layer
Model Garden
Synthetic Data Generation
Data Preprocessing
Observability Layer
Metrics Monitoring
Conversation Monitoring
Responsible AI Evaluation
Real Life Human Feedback
RAG/Model Evaluation

GenAI Reference Architecture: Patterns & Technical Blueprint for Building GenAI Solutions



Reference Architecture

Suitable for on-premise or cloud deployment.

Flexibility caters to specific organisational

needs and constraints,

regardless of infrastructure.

Reference Architecture

Same architecture

can be used in

a datacenter or

the cloud without

significant changes.

Reference Architecture

Benefits include

the choice of deployment environment,

control and security for

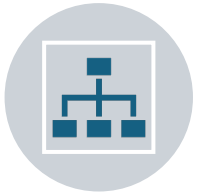
on-premise deployment,

scalability of the cloud,

or a hybrid of both.

Reference Architecture

Depending on use cases and



the organisational governance standards,

components can be moved between

the cloud and on premise

environments as required.

Reference Architecture

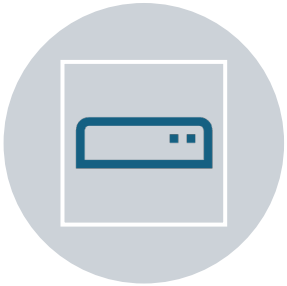
The architecture acts as a

blueprint for a generative AI system,

accommodating different AI maturity levels and

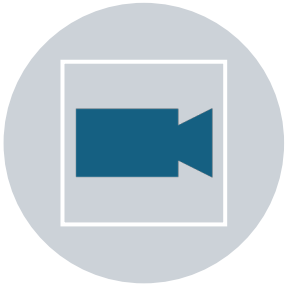
adaptable to evolving AI capabilities and needs.

Layers & Components



Integration layer

Agent Layer



RAG Layer

Model Layer

Layers & Components

MLOps Layer

Existing Data Layer

Tuning Layer

Observability Layer

1. Integration Layer

Purpose:

Handles communication between

agents and

external/internal systems.

1. Integration Layer

Components:

Internal/External Services:

Systems interacting with

the AI (data input/output).

1. Integration Layer

Components:

Event-Driven Agent Communication:

Pub/Sub architecture (e.g., Kafka)

For autonomous,

scalable agent communication.

2. Agent Layer

PURPOSE:

EXECUTES AGENT LOGIC &

ORCHESTRATES TASKS.

2. Agent Layer

Components:

Agent Orchestration:

Standardized framework

for agent structure

(prompts, RAG tools, permissions).

2. Agent Layer

Components:

Prompt Engineering:

Iterative prompt optimization;

includes guardrails and

adversarial testing.

2. Agent Layer

COMPONENTS:

MEMORY MANAGEMENT:

SESSION-BASED OR LONG-TERM MEMORY;

OPTIONAL DATA MASKING FOR SENSITIVE USE CASES.

3. RAG Layer

PURPOSE:

ENHANCES CONTEXT AWARENESS

VIA RETRIEVAL AUGMENTED GENERATION.

3. RAG Layer

Components:

Knowledge Sources:

Vector DBs,

SQL/NoSQL databases,

knowledge graphs, APIs.

4. Model Layer

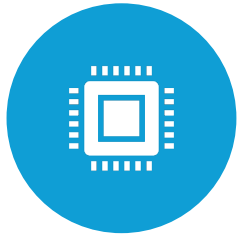
PURPOSE:

INTEGRATES LLMS AND EMBEDDING MODELS.

4. Model Layer

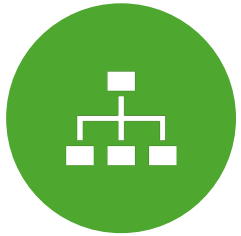
COMPONENTS:

MODEL GATEWAY:



STATELESS HTTP/HTTPS CALLS FOR MODEL ACCESS;

SUPPORTS LOAD BALANCING AND



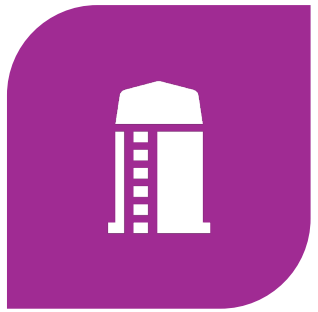
VERSION MANAGEMENT.

5. MLOps Layer

PURPOSE

AUTOMATES TESTING, DEPLOYMENT

CI/CD FOR MODELS AND



RAG PIPELINES.

5. MLOps Layer

COMPONENTS:

RAG PIPELINE:

CONVERTS SOURCE DATA INTO CHUNKS,

INDEXES VECTOR DB,

EVALUATES RELEVANCE.

5. MLOps Layer

COMPONENTS:

MODEL DEPLOYMENT PIPELINE:

CI/CD FOR DEPLOYING AND

MANAGING LLM VERSIONS.

5. MLOps Layer

COMPONENTS:

MODEL FINETUNING PIPELINE:

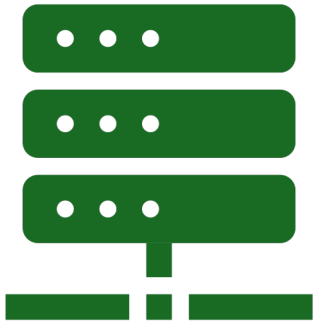
AUTOMATES TRAINING,

BENCHMARKING, AND

EVALUATION OF FINETUNED MODELS.

6. Existing Data Layer

Purpose:



Provides raw data from organizational sources.

6. Existing Data Layer

Components:

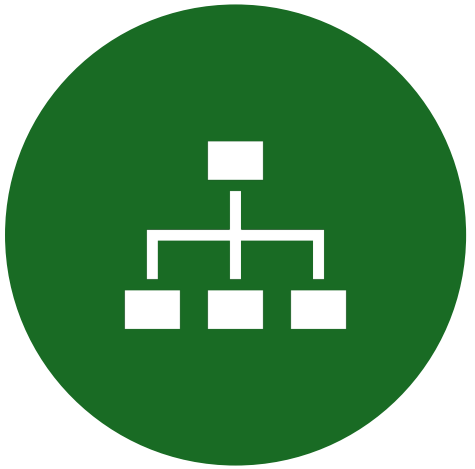
Structured/unstructured repositories

(data lakes, DBs, streams).

Supplies transactional,

customer, and real-time data.

7. Tuning Layer



PURPOSE:

CENTRAL HUB FOR MODEL TRAINING AND FINETUNING.

7. Tuning Layer

COMPONENTS

MODEL GARDEN

REPOSITORY FOR BASE AND FINETUNED MODELS.

7. Tuning Layer

Components:

Data sources:

Existing Data Layer + Observability feedback.

Data masking for sensitive info;

synthetic data generation.

8. Observability Layer

PURPOSE:

MONITORS PERFORMANCE,

SAFETY, AND USER FEEDBACK.

8. Observability Layer

Components

Metrics Monitoring:

TTFT, response time,

token usage,

cost optimization.

8. Observability Layer

Components

Conversation Monitoring:

Detect sensitive data,

harmful behavior,

hallucinations.

8. Observability Layer

REAL LIFE HUMAN FEEDBACK:

INTERNAL AND END-USER FEEDBACK LOOPS

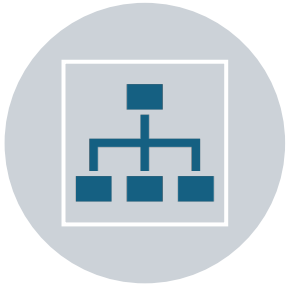
(E.G., THUMBS UP/DOWN).

Cloud Architecture for Public Data

Includes all components of the reference architecture.

Leverages **cloud scalability and flexibility** for varying workloads.

Best suited for **public/non-sensitive data**.



Enables **on-demand resource allocation** for efficiency.

Level 3 Maturity Flow (Cloud)

External service → **Integration Layer** → **Agent Layer**.

Agent interacts with **Model Layer** via Model Gateway

Model calls **RAG Layer** for context enrichment.

Level 3 Maturity Flow (Cloud)

Response flows back:

RAG → Agent → Integration → External service.

Enterprise data + feedback used for

LLM finetuning and **vector DB indexing**.

On-Premise Architecture for Sensitive Data

Same components as

Reference architecture with **enhanced security:**

Access controls, encryption, security audits.

On-Premise Architecture for Sensitive Data

INTEGRATES WITH **HYBRID CLOUD** FOR FLEXIBILITY.

SCALABILITY VIA **KUBERNETES**

On-Premise Architecture for Sensitive Data

Scalability via **Kubernetes**

Storage alternatives:

MongoDB for unstructured data.

Higher reliance on **self-hosted LLMs**

robust finetuning pipeline needed.

Hybrid Cloud Architecture Considerations

Combines

on-premise security

with **cloud scalability**.

Sensitive data stays **on-premise**;

compute and hosting in **cloud**.

Hybrid Cloud Architecture Considerations

Private connections

AWS Direct Connect

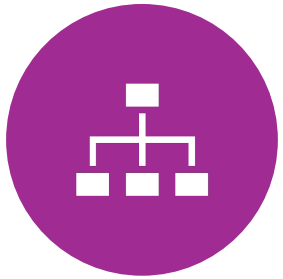
for secure traffic.

Hybrid Cloud Architecture Considerations

Balanced approach for

cost, security, and efficiency.

Final architecture depends on



business priorities and use cases.

Business Objectives and Constraints

Pharma Compliance Requirements

Regulatory Frameworks

Pharma companies follow strict regulations like HIPAA, GxP, and 21 CFR Part 11 to ensure data security and system validation.

Compliance by Design

AI components are built to comply from data ingestion to model inference, ensuring security and audit readiness.

.

Pharma Compliance Requirements

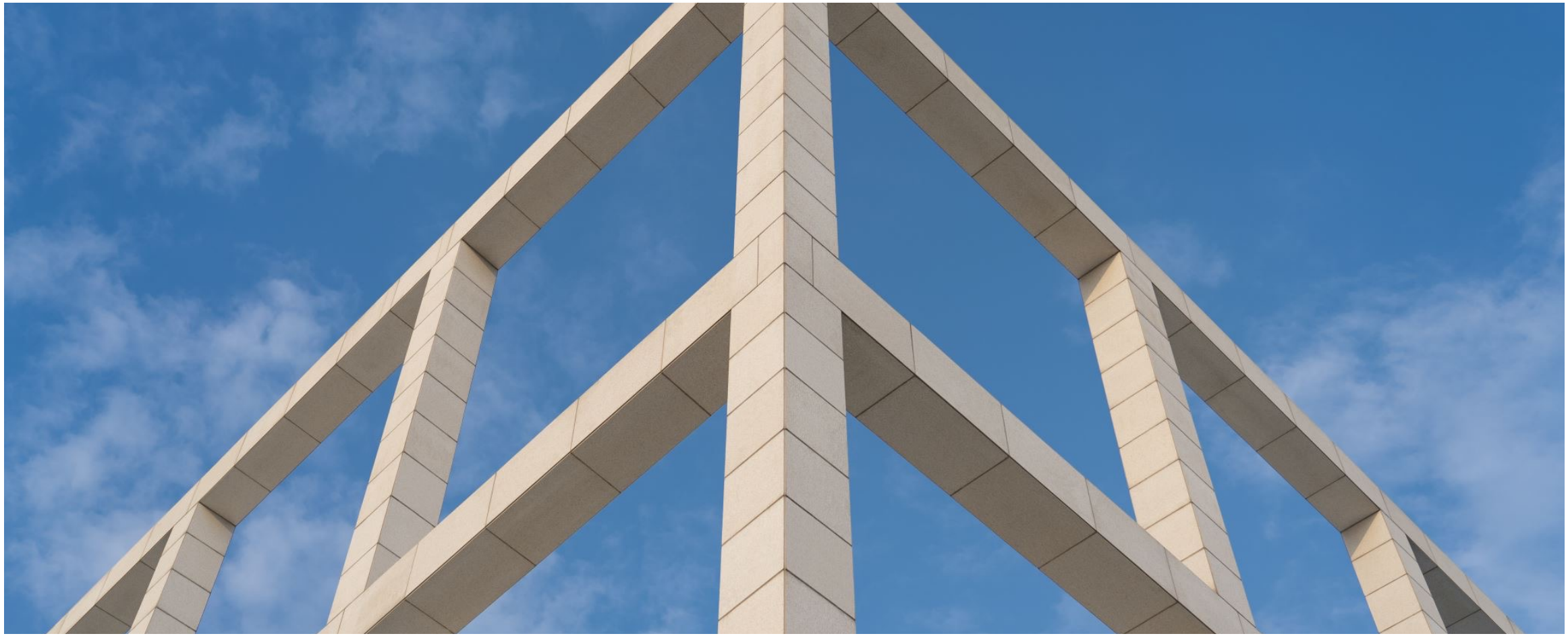
Key Compliance Features

Features like PHI redaction, role-based access control, and validation protocols maintain regulatory standards in pharma.

Data Integrity Principles

ALCOA+ principles ensure data integrity and traceability, supporting audits and quality assurance.

Phased Reference Architecture



Overview of Phases 0 to 5

Phase 0: Foundational Setup

Focus on policy, risk management, and establishing foundational infrastructure for AI deployment.

Phase 1: Core Platform Building

Development of core platform components including private large language model endpoints and vector databases.

Overview of Phases 0 to 5

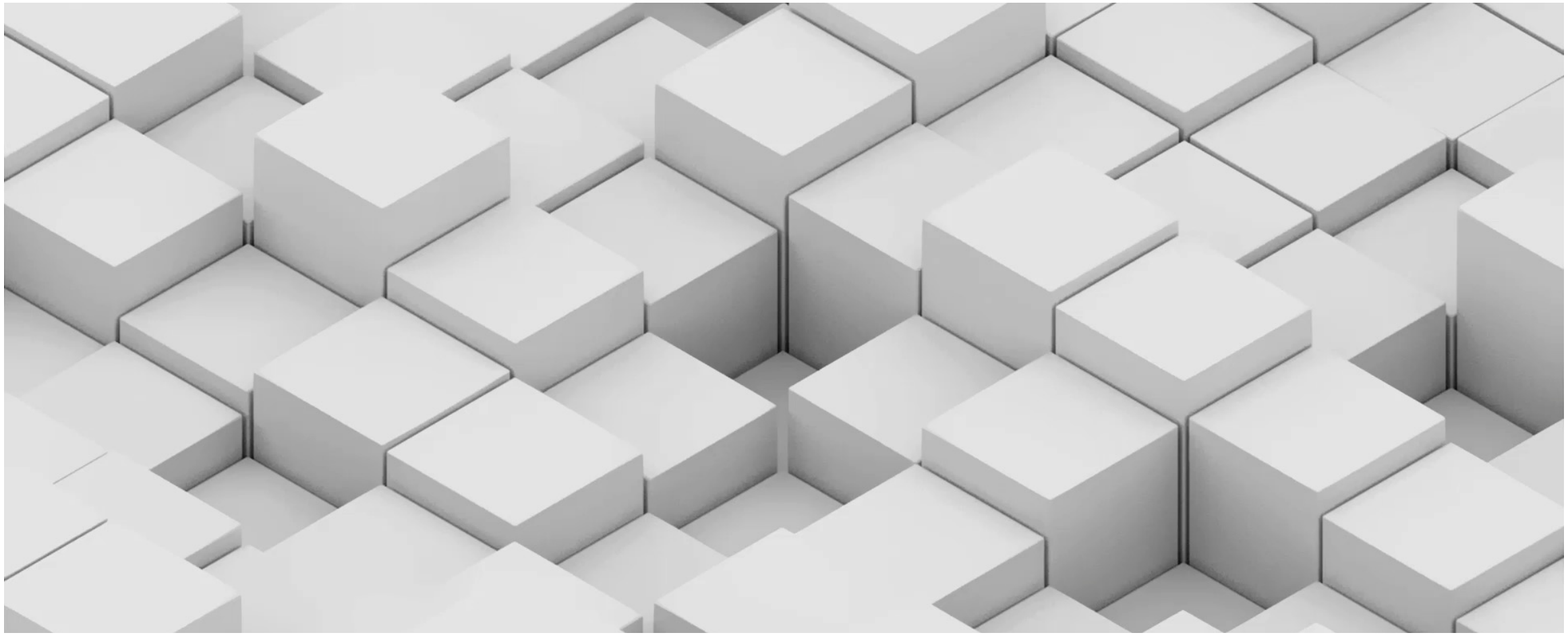
Phase 2: Agentic Workflows

Introduction of domain-specific agents to support workflows in pharmacovigilance, quality assurance, and compliance.

Phase 3 to 5: Validation and Scaling

Validation protocols, operational monitoring, canary deployments, and continuous improvement through feedback loops.

Architecture Layers



Data & Knowledge Layer Options

OPTION	COST	PERF. (HYBRID SEARCH)	ENTERPRISE FEATURES	SCALABILITY	NOTES
PGVector (Postgres)	₹	★★★★☆	Row/column security, SQL skills reused	Vertical + read replicas	Great for regulated orgs; ops-simple
Azure Cosmos DB + Vector	₹₹	★★★★☆	Managed, RBAC, regional	Elastic	Pairs well with Azure stack

Data & Knowledge Layer Options

OPTION	COST	PERF. (HYBRID SEARCH)	ENTERPRISE FEATURES	SCALABILITY	NOTES
Milvus	₹–₹₹	★★★★☆	Powerful ANN; needs ops maturity	Horizontal	High-scale vector workloads
Chroma	₹	★★★	Simple; good for POC	App-scale	Lightweight, fast POCs

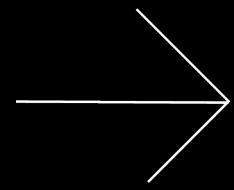
Model Layer Options

OPTION	COST (RELATIVE)	PERFORMANCE	FEATURES	SCALABILITY	DATA RESIDENCY & PHI
Azure OpenAI (GPT-4/5)	₹₹₹	★★★★☆	Function calling, long context, private endpoints	Azure scale, regional	Strong: enterprise, private; aligns with M365/Copilot
OpenAI (Enterprise / GPT-5)	₹₹₹	★★★★★	Latest reasoning features; caching	Global API scale	Enterprise controls; use with private networking

Model Layer Options

OPTION	COST (RELATIVE)	PERFORMANCE	FEATURES	SCALABILITY	DATA RESIDENCY & PHI
Amazon Bedrock (Claude / Titan)	₹₹–₹₹₹	★★★★☆	Guardrails, model choice	AWS scale	VPC endpoints, IAM integration
Self-host (Llama-3/4, Mixtral)	₹–₹₹	Varies	Full control; on-prem	Kubernetes scale; hardware bound	Full PHI control; higher ops burden

Agentic AI Frameworks

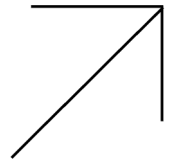


Framework Options and Use Cases

FRAMEWORK	COST	STRENGTHS	BEST USE	MATURITY
LangGraph	₹	Stateful graphs, retries, checkpoints	Regulated, multi-step flows (PV/QA)	Prod-ready patterns
LangChain	₹	Tooling ecosystem, retrievers	RAG pipelines + tools	Widely adopted
Semantic Kernel	₹	.NET, M365 skills, planners	Deep Microsoft enterprise integration	Strong for M365

Framework Options and Use Cases

FRAMEWORK	COST	STRENGTHS	BEST USE	MATURITY
Autogen	₹	Multi-agent prototyping	Early ideation, experiments	Proto → formalize later
Microsoft Copilot Studio	₹₹	Citizen dev, connectors	Simple line-of-business bots	Business-led scenarios

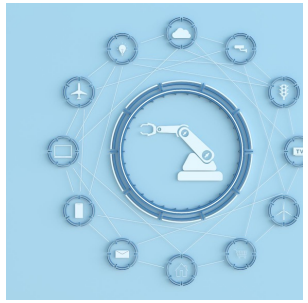


Validation and Observability

Validation and Audit Readiness

Validation Architecture

Validation includes IQ, OQ, and PQ to ensure AI system setup, functionality, and performance meet standards.



Continuous Integration and Deployment

CI/CD pipelines automate validations, generate reports, and support change control board approvals.

Regression Gates and Metrics

Regression gates require high grounded accuracy and citation coverage before AI production release.

Observability and Governance

DOMAIN	OPTIONS	WHY IT MATTERS IN PHARMA
Tracing/Evals	LangSmith, custom eval harness, Azure ML evals	Repro, defect triage, grounded accuracy KPIs
Monitoring	Azure Monitor, Prometheus/Grafana, SIEM	Latency, tokens, refusal, safety events

Observability and Governance

DOMAIN	OPTIONS	WHY IT MATTERS IN PHARMA
Catalog/Lineage	Purview, model registry, doc registry	ALCOA+ evidence, 21 CFR Part 11 trails
Policy & Safety	Prompt guardrails, PHI redaction, deny-list	HIPAA/GxP; “cite-or-don’t-say” enforcement

Deployment and Cost Considerations

Deployment Topologies

DEPLOYMENT	TYPICAL USE	CONTROLS	PROS	CONS
Azure (AKS/ACA + Private Endpoints)	Most Viatris workloads	Key Vault, Private Link, Entra ID	Native with Copilot/M365	Cloud residency constraints
AWS (EKS + Bedrock)	AWS-centric teams	VPC, KMS, IAM	Broad model palette	M365 integration indirect

Deployment Topologies

DEPLOYMENT	TYPICAL USE	CONTROLS	PROS	CONS
On-prem / Air-gapped (K8s)	PHI-heavy, PQ validation	HSM, zero egress	Full control	Hardware + ops overhead
Hybrid	Mixed sensitivity	Private connectivity	Right-sizing risk & cost	Network design complexity

Cost-Performance Comparison

Enterprise AI Features

Azure OpenAI provides strong enterprise features with higher cost, suitable for robust, secure deployments.

Cost-Effective Vector Storage

PGVector offers affordable vector storage with strong governance for efficient data management.

Cost-Performance Comparison

Scalable AI Workloads

Bedrock and Milvus support scalable AI workloads for AWS-centric and high-volume applications.

Specialized Deployment Scenarios

Self-hosted models and Chroma suit air-gapped environments and proof-of-concept testing effectively.

Roll-out Strategy and Performance Metrics

Phased Rollout Approach

The strategy moves from pilot to full deployment in phases to ensure smooth implementation and scalability.

Key Performance Indicators

KPIs like accuracy above 90%, citation coverage above 95%, and low latency ensure system effectiveness.

Roll-out Strategy and Performance Metrics

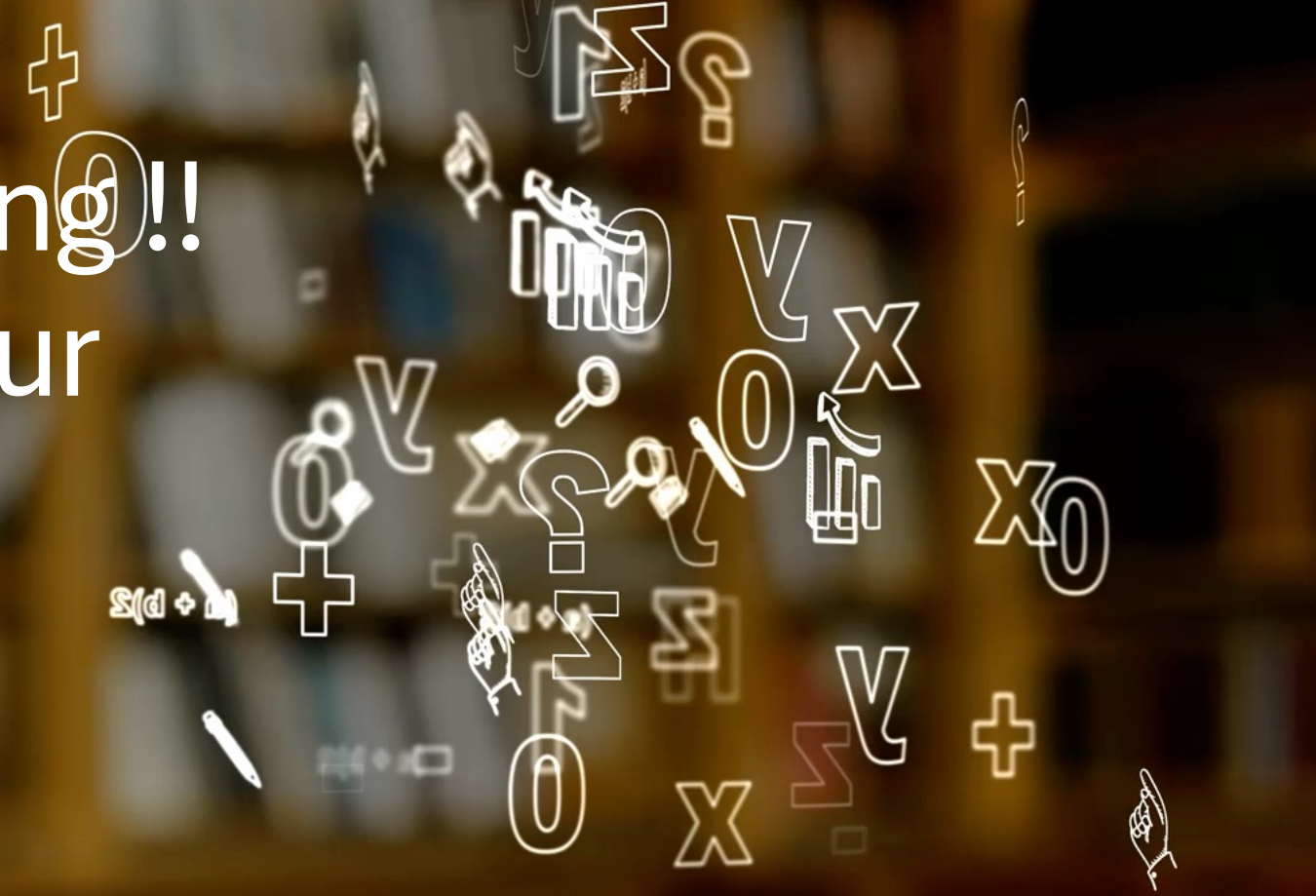
Compliance and Security

Zero PHI leakage and ALCOA+ compliance with audit readiness at 95% maintain data security and integrity.

Efficiency Improvements

PV documentation time is reduced by 60–70% while maintaining human approval for quality control.

Happy Learning!! Thanks for Your Patience ☺



Surendra Panpaliya

GKTCS Innovations