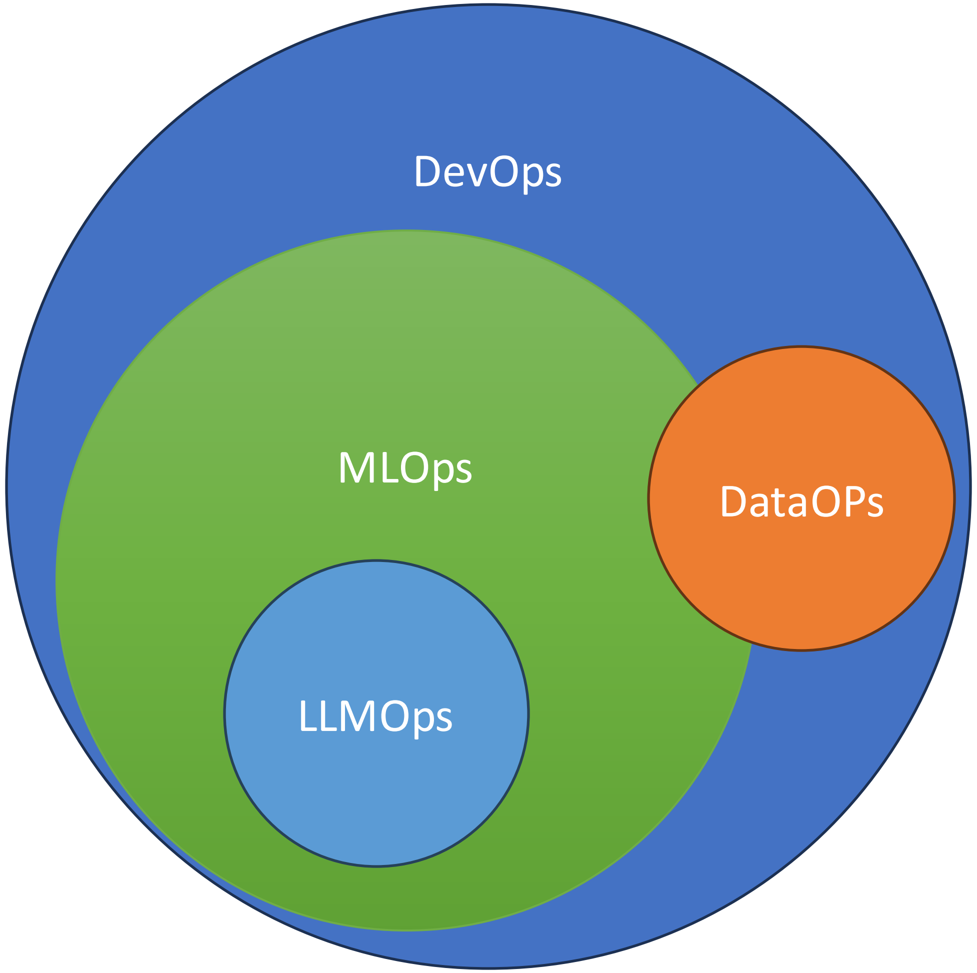
Large Language Models Operations (LLMOps)

Content

- LLMOPs introduction
- Deployment and scalability of LLMs
- Monitoring and maintenance of models in production
- Performance evaluation and continuous improvement
- Ethical considerations and privacy

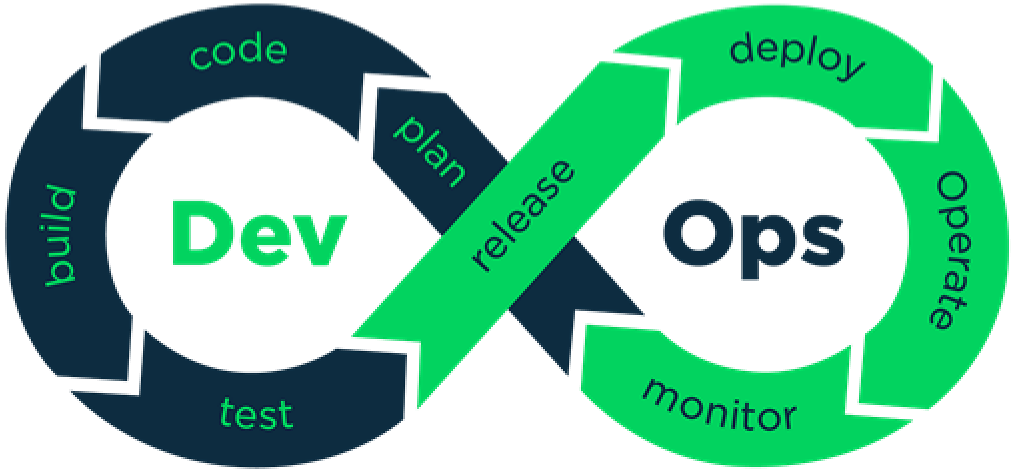
Intro to LLMOPS

XOPs

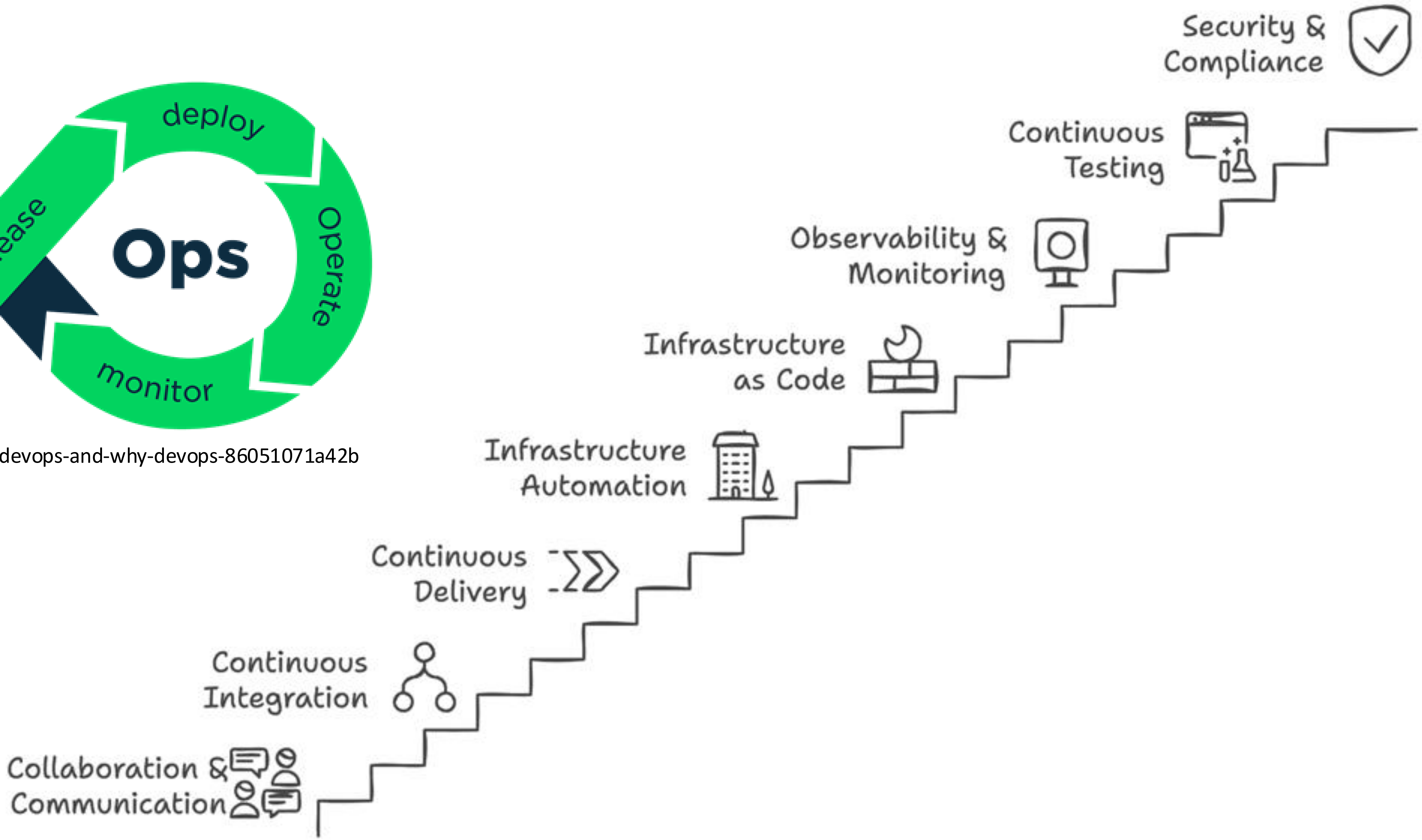


- Automation
- Collaboration
- Reproducibility
- CI/CD
- Observability

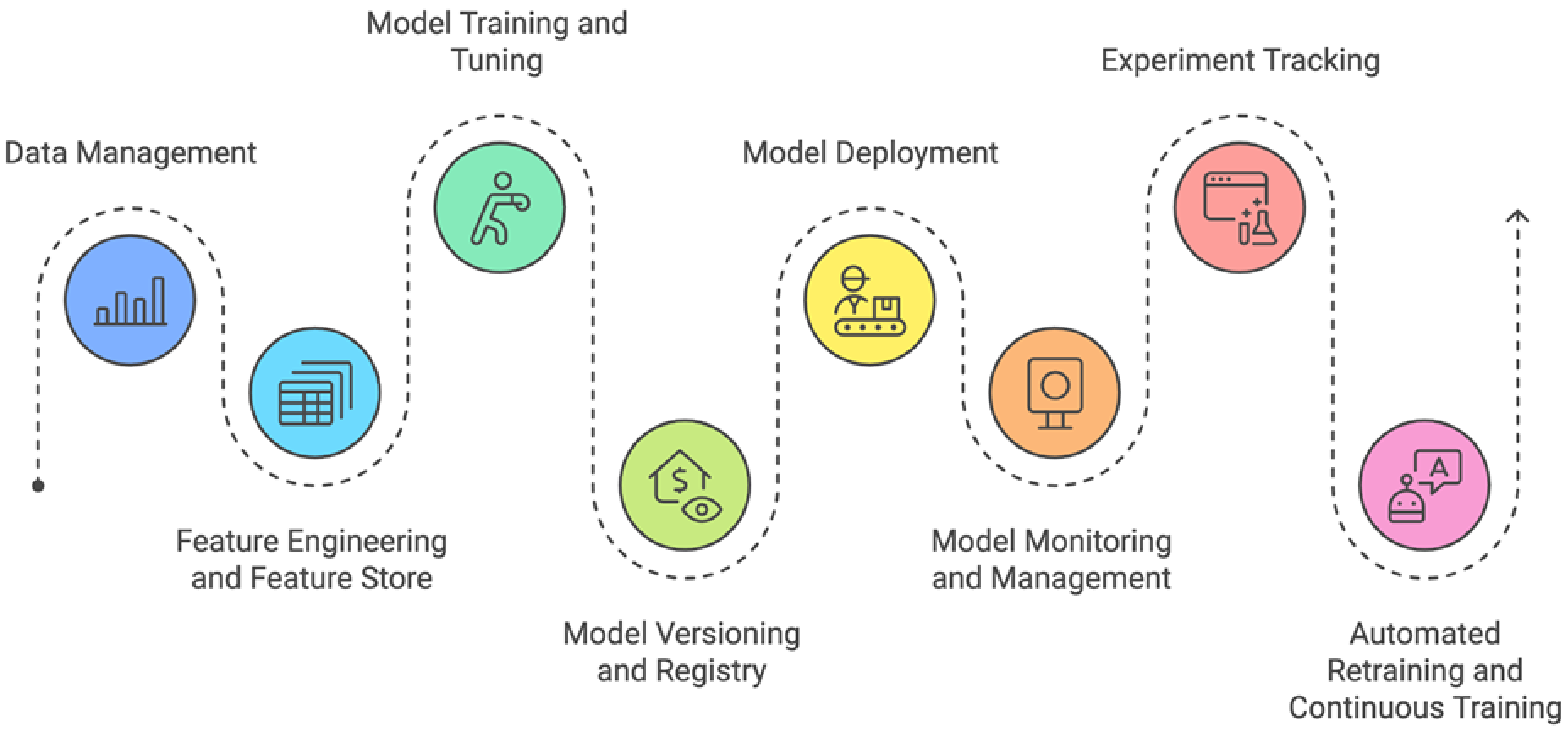
DevOps



https://medium.com/@ritusherke86/what-is-devops-and-why-devops-86051071a42b



MLOps



Model Training and Tuning
Experiment Tracking
Data Management
Model Deployment
Feature Engineering and Feature Store
Model Monitoring and Management
Model Versioning and Registry
Automated Retraining and Continuous Training

LLMOps

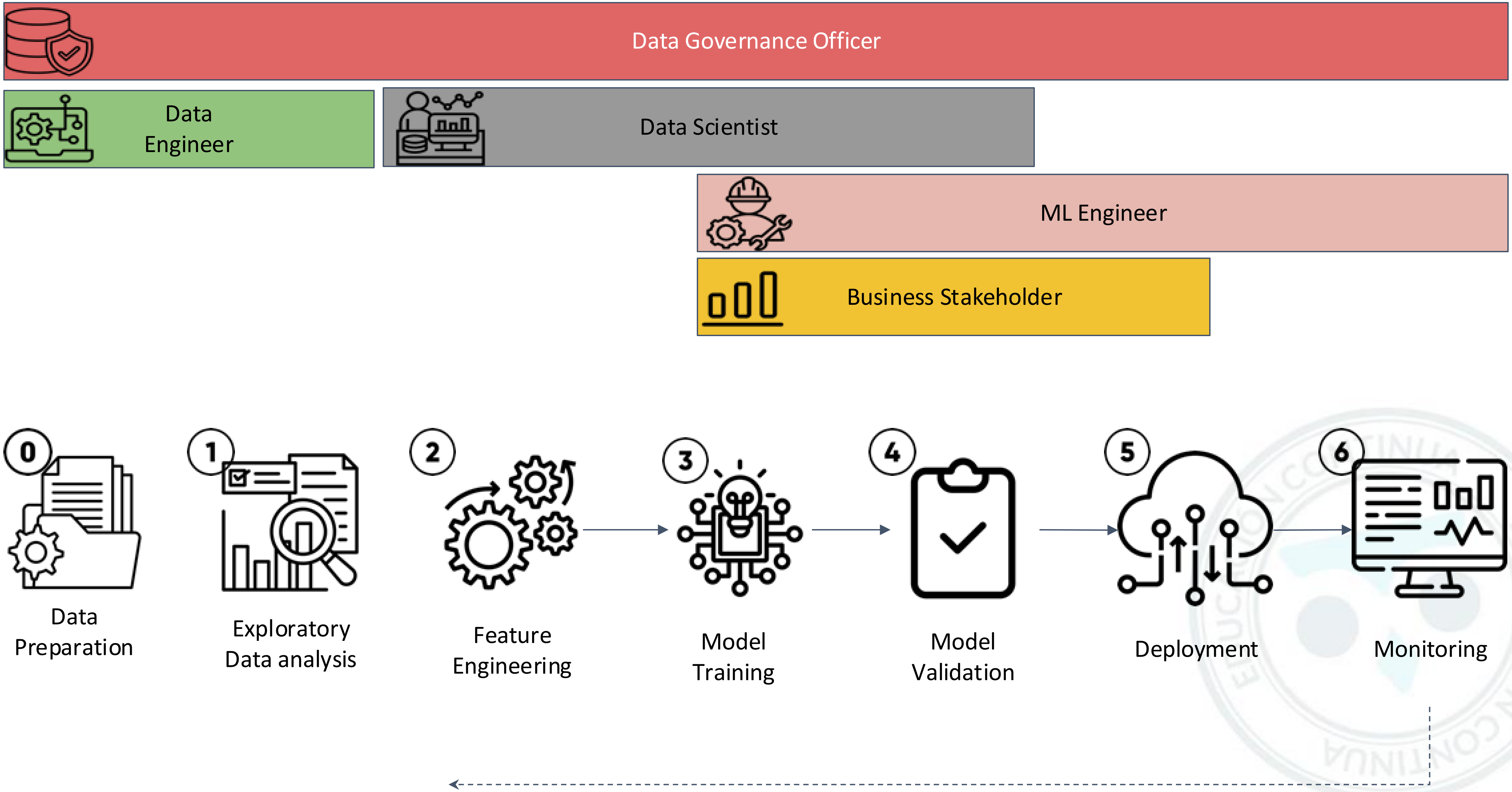

Resource Optimization
Prompt Engineering
Bias Detection and Mitigation
Database Management
Human Feedback Integration
LLM Architecture
LLM Evaluation Metrics

UNIVERSIDAD EIA

EDUCACIÓN CONTINUA

Training and deployment of Large Language Models

MLOps


Data Governance Officer
Data Engineer
Data Scientist
ML Engineer
Business Stakeholder
0
Data Preparation
1
Exploratory Data analysis
2
Feature Engineering
3
Model Training
4
Model Validation
5
Deployment
6
Monitoring

What specific scaling challenges exist for LLMs?

- **Initial training**: trillions of tokens, hundreds to thousands of GBs & very long run times.
- **Fine-tuning**: updating model weights based on your own data, still requires relatively large data and long training times. Plus lots of evaluation!
- **Storage**: the models contain billions of parameters, often hundreds of GB. This can be prohibitive for some devices.
- **Latency**: Running inference on these models can be very costly in terms of time.
- **Cost**: All of this costs $$$!

MLOps

What specific scaling challenges exist for LLMs?

Initial training: don't do it!

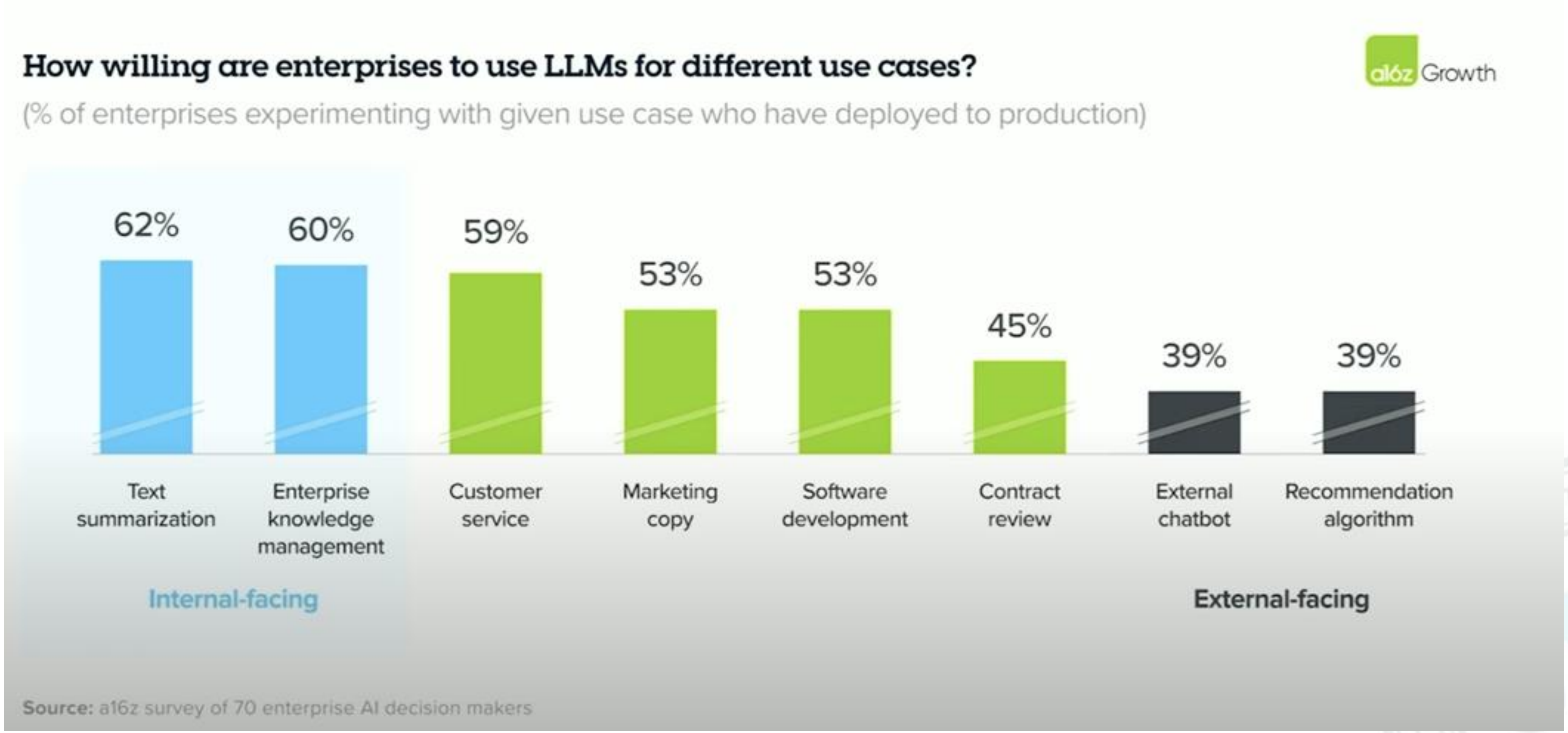
Fine-tuning: Optimize and use a scalable framework, such as Ray.

Storage: Quantization, memorization, caching …

Latency: Quantization, memorization, caching, hardware and memory bandwidth optimization…

Cost: Above plus use 'open source' models

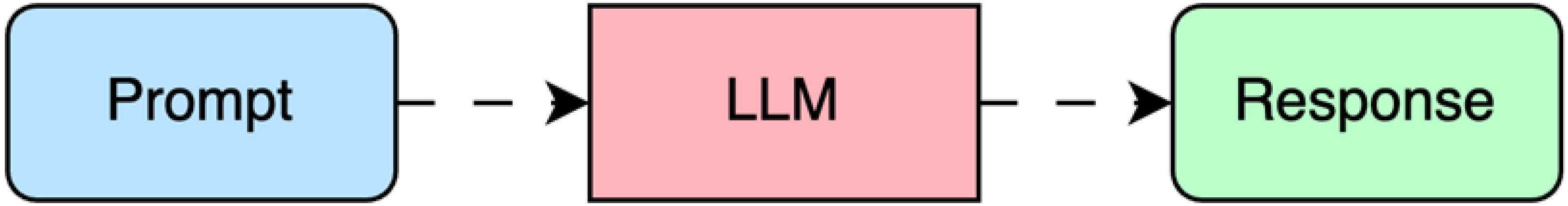
Types of use-cases for Enterprises

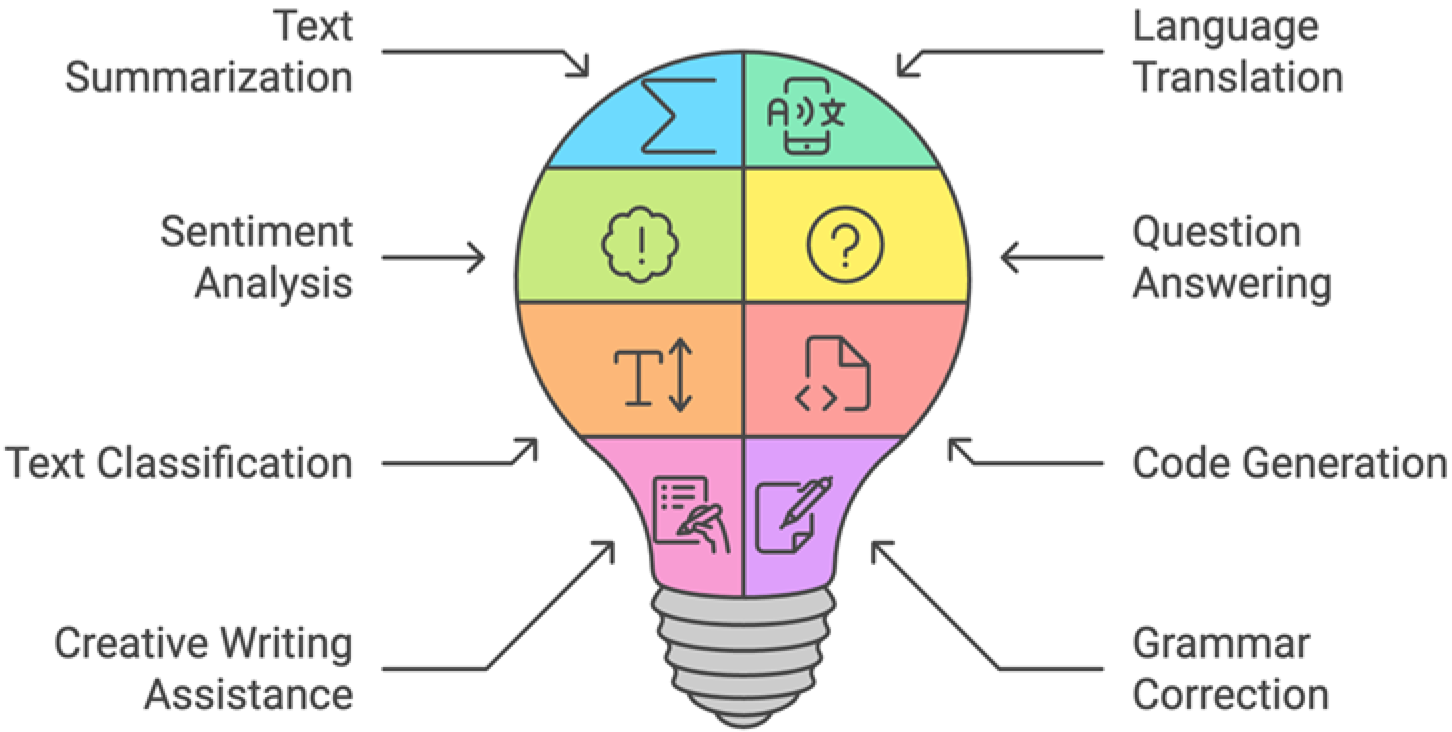


LLM application archetypes

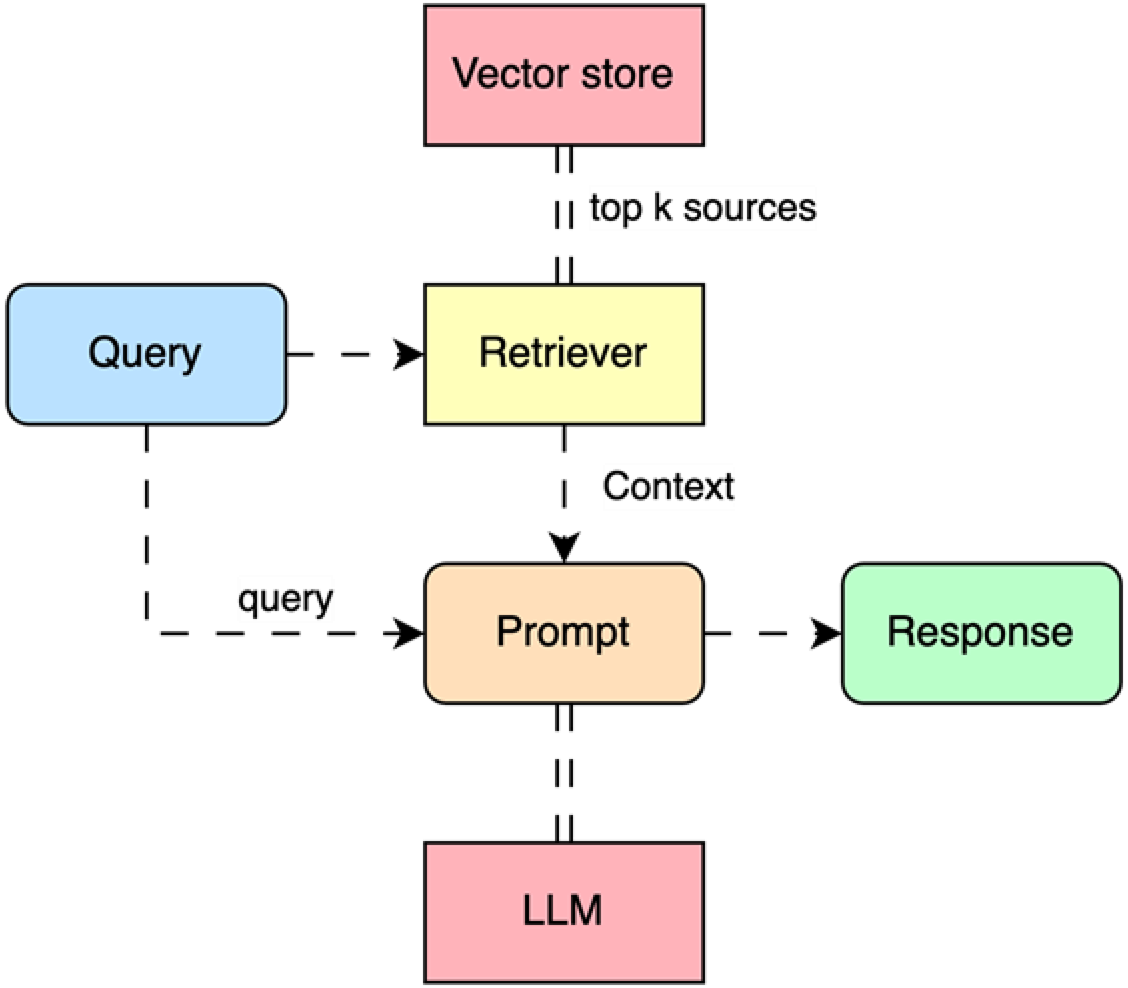
- Prompt Engineering
- Retrieval Augmented Generation
- Agents
- Multi-Agents
- Fine-tunning

Prompt Engineering Applications

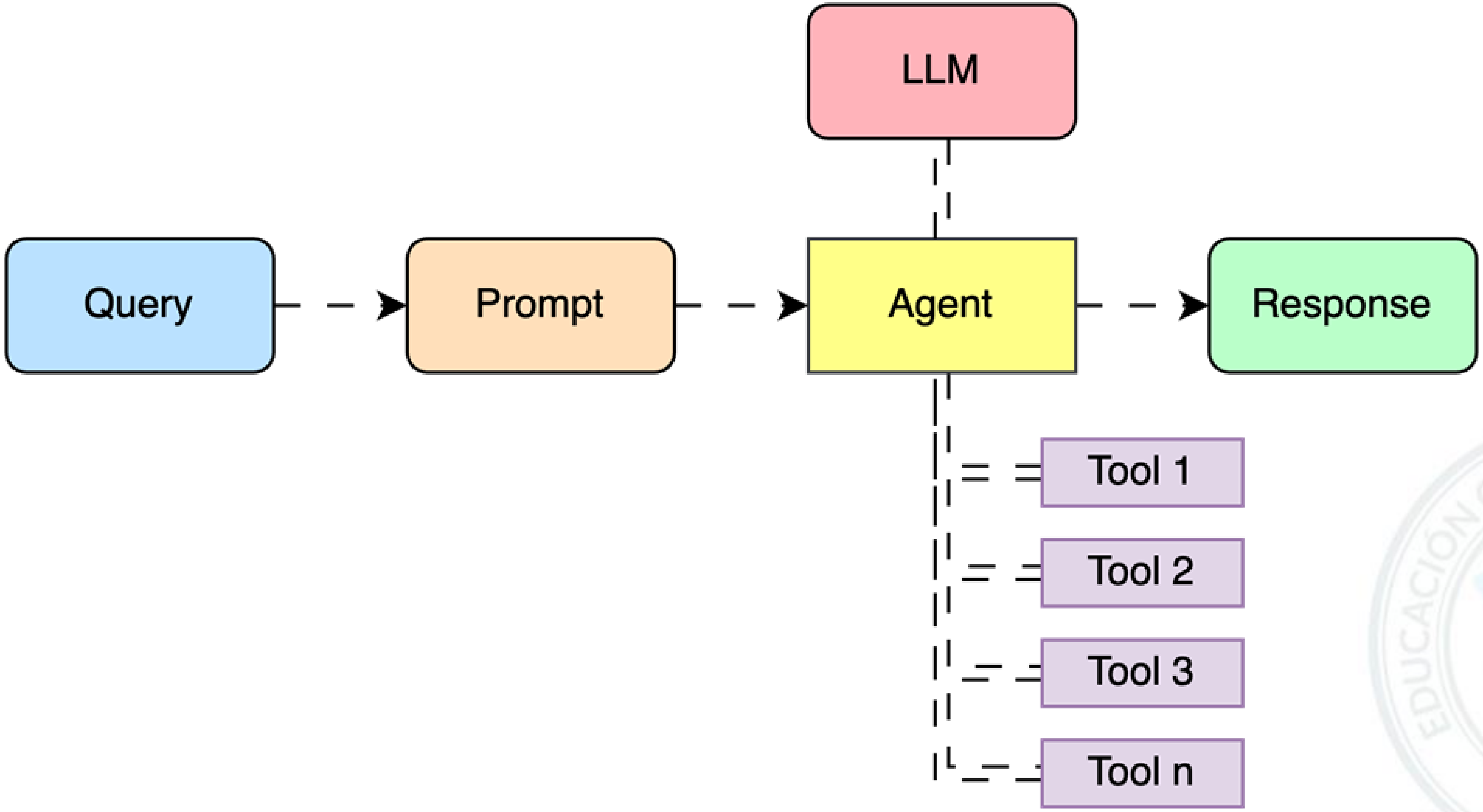




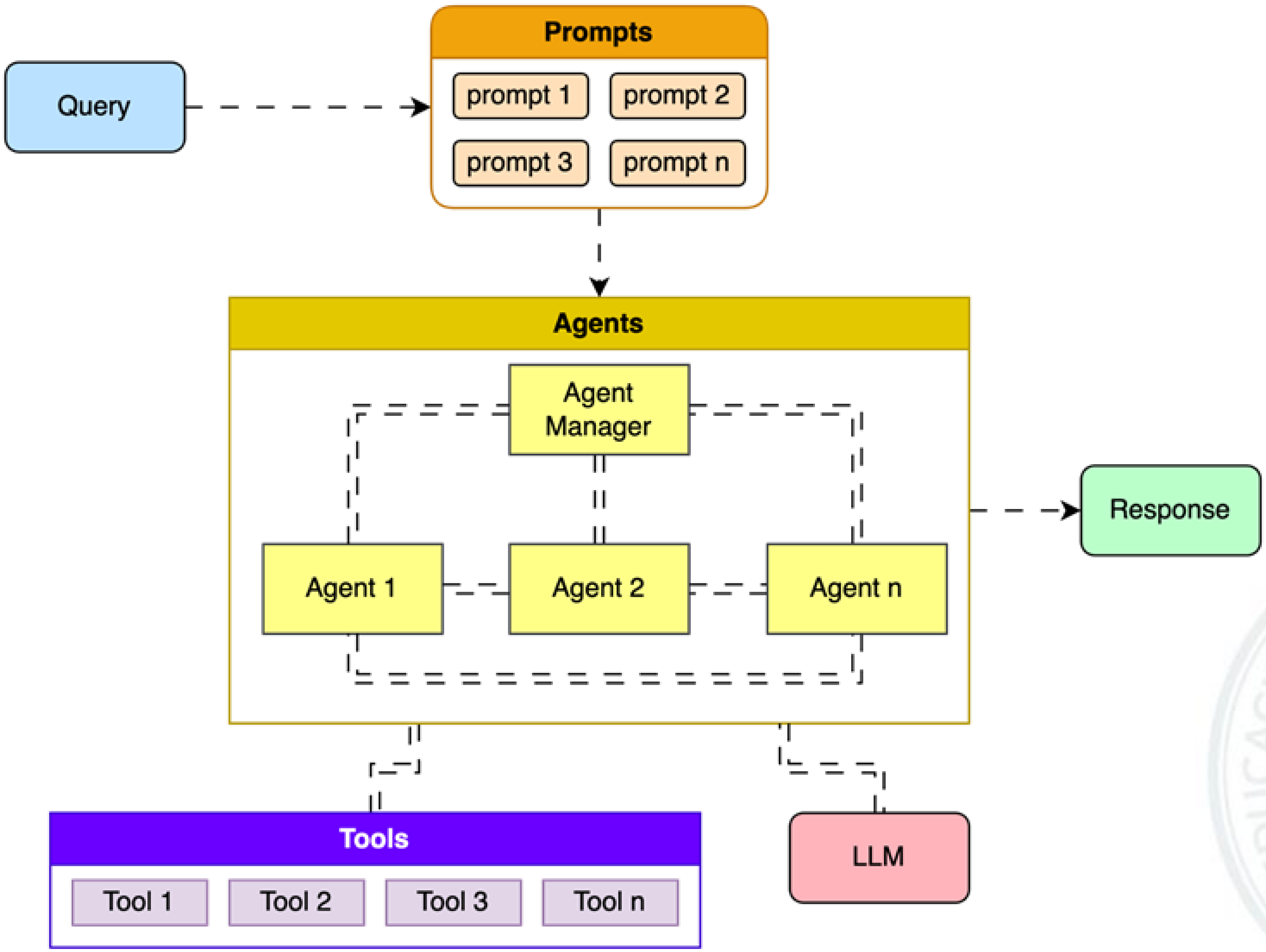
Retrieval Augmented Generation Applications



Agentic Applications



Multi-Agent Applications



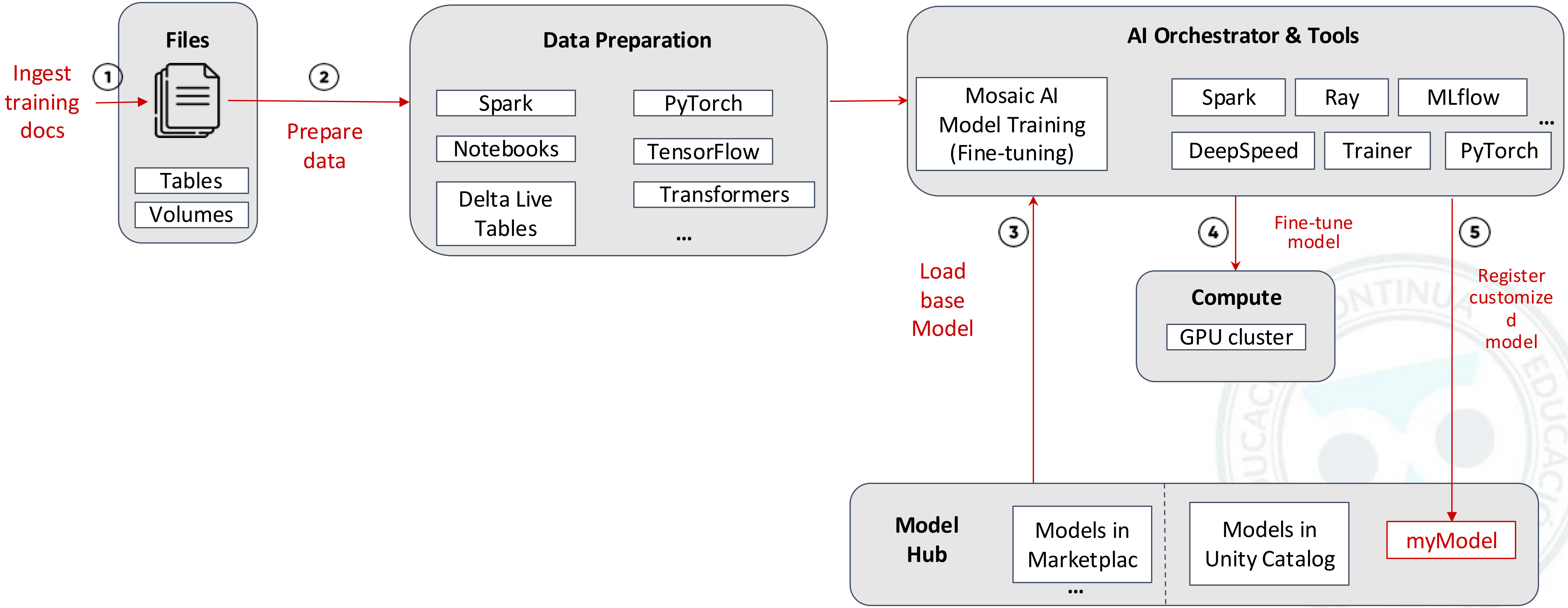
Fine-tunning LLM Applications

Operationalizing Fine-tuning Pipelines

- Automation, version control, reproducibility
- Distributed training infrastructure
  - DeepSpeed, PEFT, GPUs
- Similar to classical model training serving
- Optimization techniques
  - vLLM, MLC, CudaGraph, MQA, Quantization, TensorRT
- Deeper understanding of GPUs, TTFT, TPOT

Types of LLM Applications

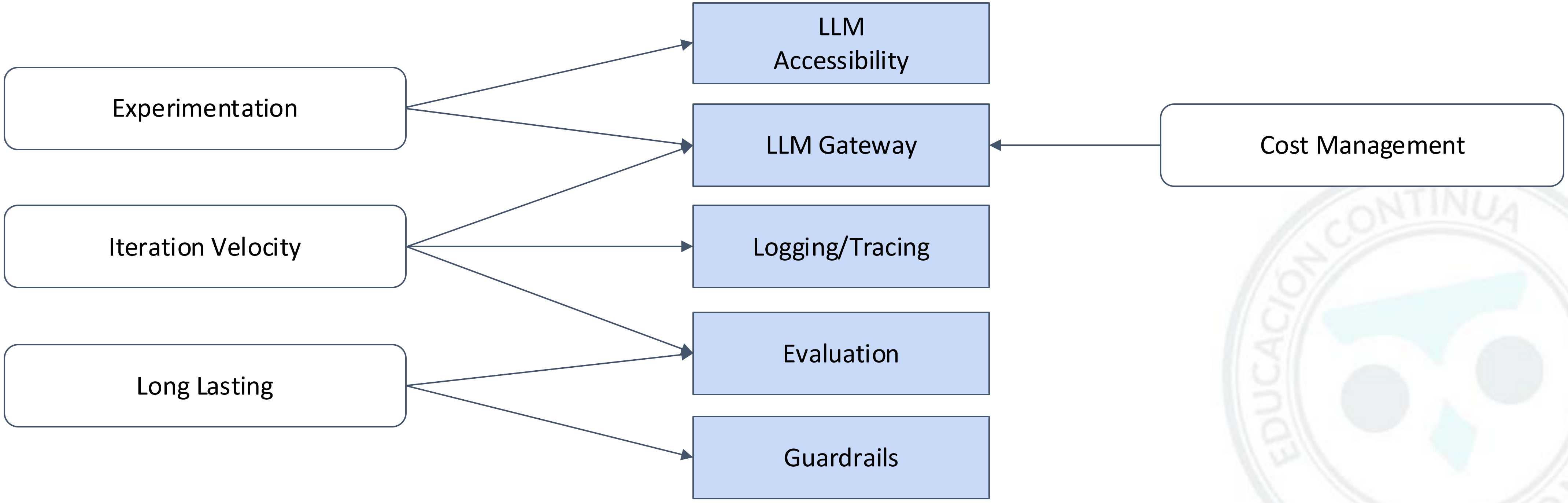
Architecture: fine-tuning



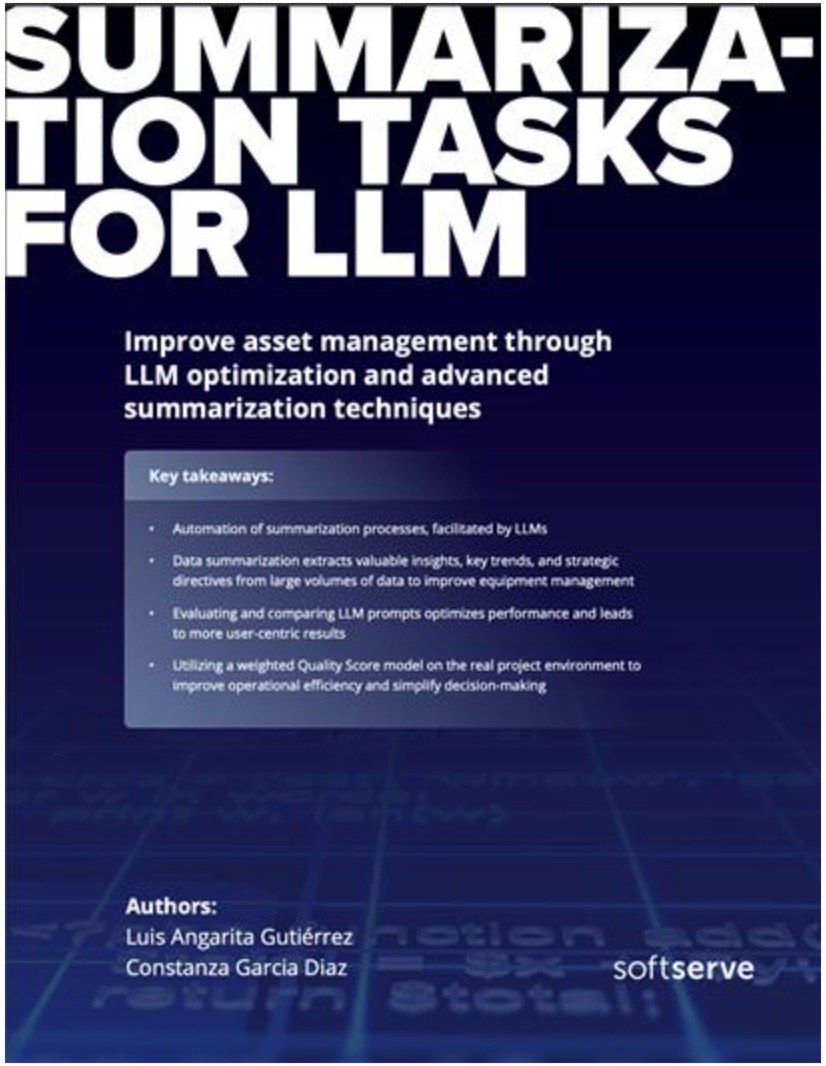
Types of LLM Applications

Practitioner's Perspective

LLMOps Starting Points



Use-Case



Improve Asset Management with LLM Evaluation Pipelines

Creating efficient AI-powered summaries, LLMs save up to 93% of the time taken by traditional manual processes. Our white paper explains how this allows you to optimize asset management operations.

https://info.softserveinc.com/summarization-tasks-for-llm-white-paper

Prompt engineering Example: Youtube Summarizer

GitHub Repo: https://github.com/LGuillermoAngaritaG/llmops-youtube-summarizer

Groq-API-Key: https://console.groq.com/keys