Workflow vs Agent-based Design

Exploring two distinct approaches for **automating Kubernetes Operator upgrades** using predictable Workflow model and adaptive Agent-based design.

by Kavinraja Ganesan

Workflow Design: Fixed K8s Operator Upgrade Pipeline

The workflow approach defines a rigid, pre-determined sequence for operator upgrades. Each step is explicitly coded, making the process highly reproducible and straightforward to debug.

Trigger & Validation

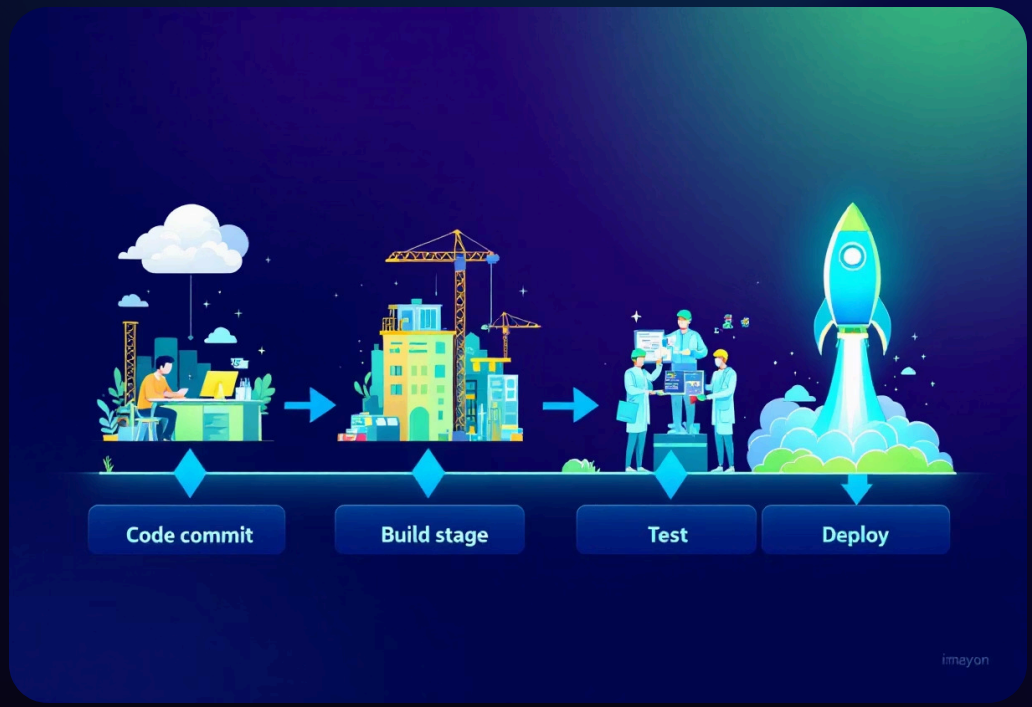
Monitors for new releases via webhooks or cron jobs, fetching changelog details.

Preflight & Execution

Backs up existing resources, then updates and applies Helm/Kustomize charts to staging.

Verification & Rollout

Runs smoke tests to ensure functionality, followed by a controlled promotion to production.



This method is ideal for routine, low-risk upgrades where the process is well-understood and rarely deviates. Error handling is typically hard-coded, ensuring predictable responses to common issues.

Agent Design: Dynamic Upgrade Agent for K8s Operators

The agent-based design uses an intelligent system to dynamically adapt its upgrade strategy. It observes, reasons, plans, and acts, making context-aware decisions.

1

Goal & Observation

Detects new operator releases and compares them against current versions.

2

Reasoning & Planning

Utilizes LLMs to analyze changelogs, CRD diffs, and simulate upgrade impact, then decides on actions.

3

Tool Use & Act

Interacts with APIs (GitHub, Kube), diff tools, validation, and PR tools to execute planned actions.

4

Replan & Dependency

If tests fail, it re-analyzes logs and proposes fallback plans, handling dependent operator upgrades.

Comparison

Feature	Workflow Based	Agent Based
Trigger Logic	Cron/GitHub event	Observed goal or user input
Tool Use	Static (Helm, kubectl, CI)	Dynamic (GitHub API, Kubectl)
Adaptability	Low	High
Failure Recovery	Manual or scripted	Built-in re-plan logic
Ideal For	Routine upgrades	Risky/complex upgrades
Auditability	High (loggable CI steps)	Medium (state must be logged)

Summary

Use Workflows when:

- Upgrade process is well-understood and predictable.
- Changes are low risk or occur frequently (e.g., weekly cert-manager updates).
- Reproducibility and consistent execution are paramount.

Use Agents when:

- Operator introduces complex CRD schema changes or breaks user CRs.
- Context-aware decisions are vital (e.g., conditional upgrades).
- Proactive reasoning and dynamic rollback strategies are critical.