



Update from the Maintainers



Lukas Reining
codecentric



Thomas Poignant
Gens de Confiance



Alexandra Oberaigner
Dynatrace



André Silva
LexisNexis Risk Solutions

Agenda

- Introduction to OpenFeature
- OFREP v1.0
- OpenFeature MCP
- OF Github Action + CLI
- Discussion & Decisions
- Q&A



What is OpenFeature?

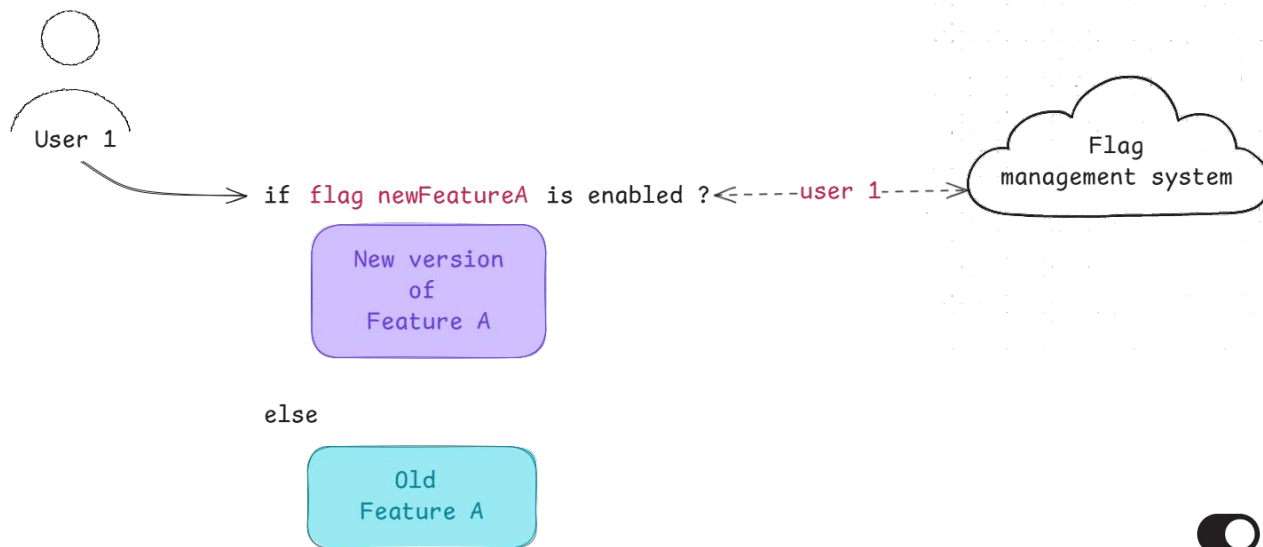
What is OpenFeature?

An **open specification** that provides a vendor-agnostic, community-driven API **for feature flagging** that works with any management tool or in-house solution.



What are feature flags?

Feature flags are a software development technique that allows teams to enable, disable or *change the behavior* of certain features or code paths in a product or service at runtime, *without modifying the source code*.



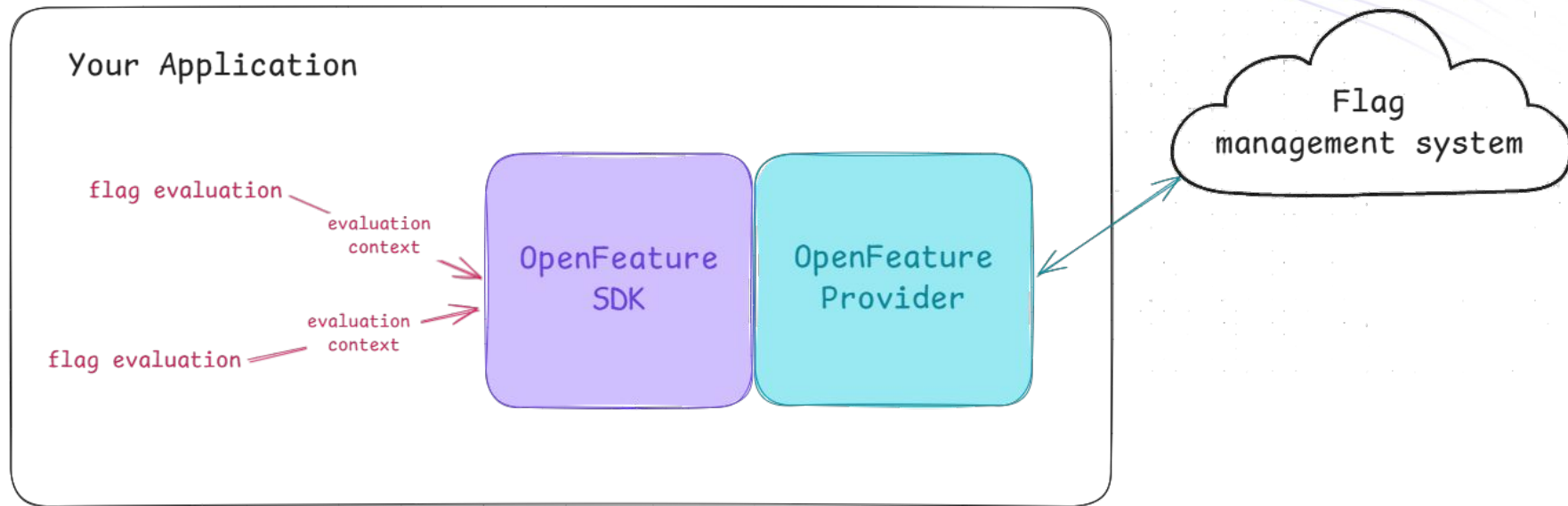
KubeCon



CloudNativeCon

Europe 2025

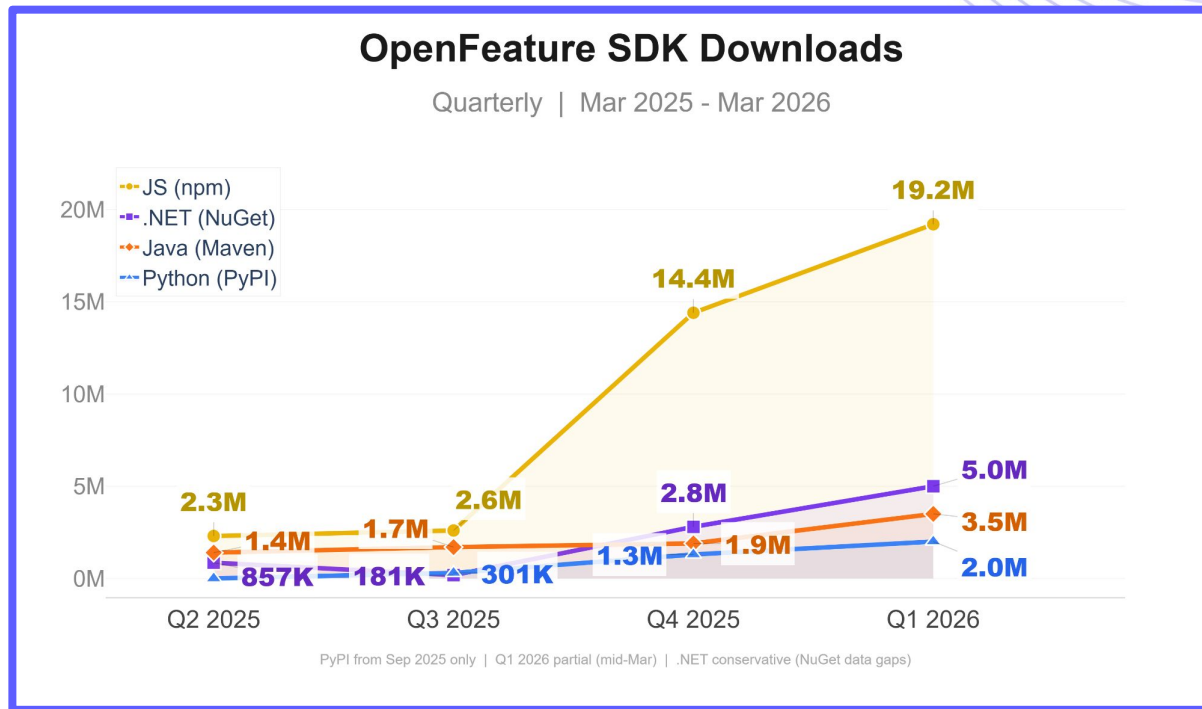
OpenFeature - Architecture



OpenFeature - SDK Landscape



OpenFeature – SDK Downloads Growth



Contributing Companies

<https://insights.linuxfoundation.org/project/openfeature/contributors>



KubeCon



CloudNativeCon

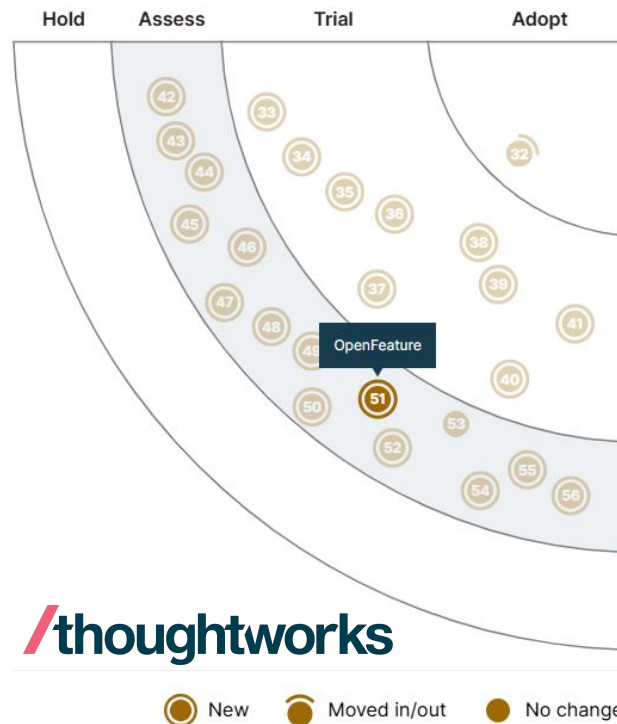
Europe 2026

#	Organization	Total contributions
1	Dynatrace, Inc.	8,240
2	LexisNexis	1,703
3	DevCycle	1,109
4	Leboncoin	887
5	codecentric AG	654
6	Exclaimer Ltd	532
7	Capital One Financial Corporation	432
8	TestifySec	355
9	Bridgecrew	326
10	Spotify AB	304
11	Gusto!	298
12	Google LLC	250
13	GLS IT Services GmbH	203
14	Booking.com	196
15	Proofpoint, Inc.	148
16	Beau Rivage Casino	130
17	VenturSeed	119
18	Datadog, Inc.	111
19	SF Civic Tech	103
20	LaunchDarkly	98

Thoughtworks Tech Radar

OpenFeature entered the Tech Radar

- Volume 33 | November 2025





What's new in OpenFeature?

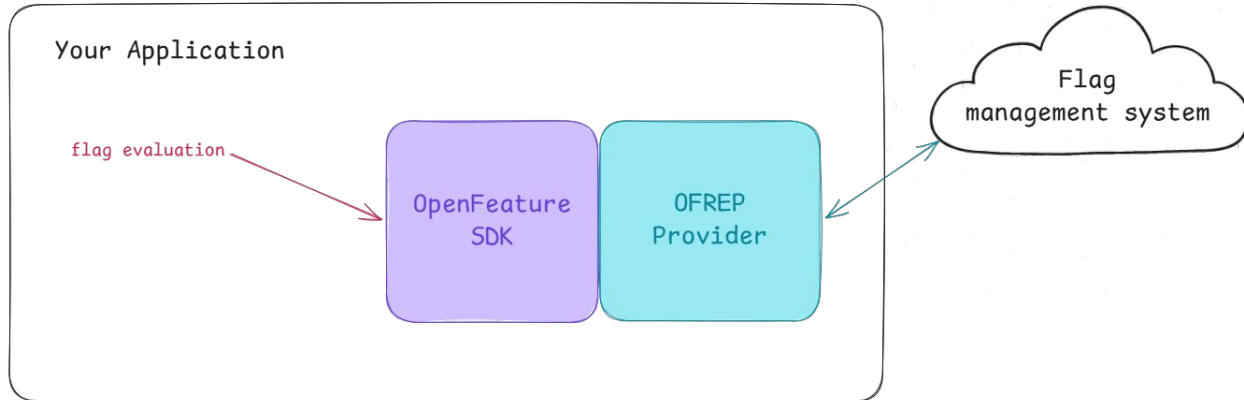


OFREP

Remote Evaluation Protocol

OpenFeature – OFREP

OpenFeature Remote Evaluation Protocol is a HTTP based protocol meant to lower the amount of provider implementations for FF vendors and tools.



OpenFeature – OFREP

OFREP Core

Required: Core APIs to implement to support OFREP.

This is the minimum set of APIs required for a flag management system to be OFREP compatible.



POST

/ofrep/v1/evaluate/flags/{key} Evaluate A Single Feature Flag



POST

/ofrep/v1/evaluate/flags Bulk Evaluate All Feature Flags



KubeCon



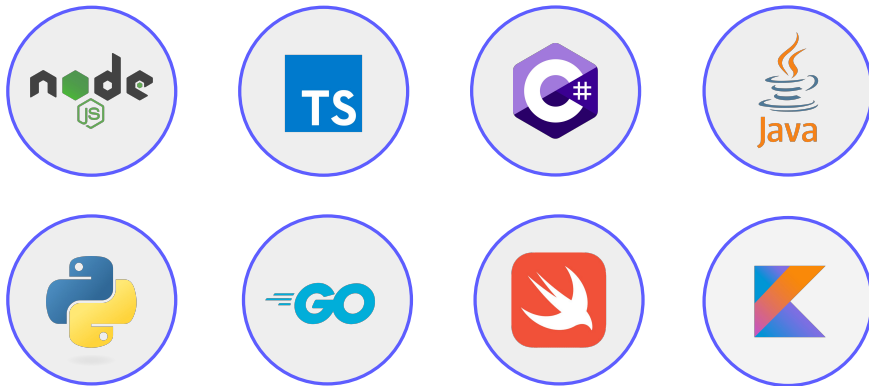
CloudNativeCon

Europe 2025

 OpenFeature

OpenFeature – OFREP

Supported OFREP providers



OpenFeature – OFREP

Benefits of using OFREP

- No Vendor-Specific SDKs or providers in your code.
- Switch flag vendors by changing a single URL, not your entire codebase.
- Community-maintained providers mean you don't have to write 10 different SDKs.

OpenFeature – OFREP

Status of OFREP

- OFREP is already adopted by multiple vendors.
- **v1.0.0 is almost out**, we've hardened the spec and providers.
- Working on the next version of OFREP with Server Sent Events support.



OpenFeature MCP

A companion for your development

OpenFeature MCP – What is it?

OpenFeature ❤️ AI Coding Assistants Tools

- Connects tools like Cursor, Claude Code, VS Code and GitHub Copilot to OpenFeature
- Enables AI agents to interact with feature flagging through a standardized protocol
- Published to the MCP Registry as **dev.openfeature/mcp**

OpenFeature MCP – What can it do?

1. SDK Installation Guidance and Intelligent Code Generation

- Fetches and provides OpenFeature SDK installation instructions
- Supports multiple technologies and frameworks
- AI-powered code generation for OpenFeature integration
- Assists with SDK migrations
- Direct integration within your development environment

2. Feature Flag Evaluation

- Evaluate flags using OpenFeature Remote Evaluation Protocol (OFREP)
- Supports both single-flag and bulk evaluation
- Bearer token and API key authentication

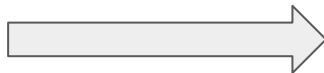
OpenFeature MCP – Examples

Add OpenFeature Go SDK



- Get SDK installation steps
- Return/write code example

AI agent queries:
"Check if feature X is enabled?"



- Call your feature flag service
- Return current flag state
- Use for intelligent deployment decisions



OpenFeature Action + CLI

Build your own flags workflow



github-actions bot commented [last week](#)



OpenFeature Manifest Comparison (validate-flags)



Warning

Git Changes Detected - Your feature flag manifest has changes compared to the main branch



Change Summary

Change Type	Count	Description
 Additions	1	Flags added in this branch
 Removals	0	Flags removed in this branch
 Modifications	0	Flags modified in this branch
Total	1	Total differences

▶  View detailed changes

▶  View manifest diff



This comparison shows changes introduced in this PR

This comment is automatically updated on each push



KubeCon



CloudNativeCon

Europe 2025

OpenFeature



github-actions bot commented [now](#)



Flag manifest is out of sync

The `flag-manifest.json` on this branch does not match what `flags-source.json` would produce.

► [View differences](#)

Check the box below to auto-regenerate:

☐ Pull updated flags and regenerate React accessors

This will run `openfeature pull` and `openfeature generate react`, then commit the result to this branch.



github-actions bot commented [18 minutes ago](#) • edited ▾



Flag manifest regenerated

`flag-manifest.json` and `src/flags/` have been updated from `flags-source.json` and committed to this branch.



☒ Pull updated flags and regenerate React accessors



KubeCon



CloudNativeCon

Europe 2026

OpenFeature

Why use it?

- Maintain Consistency
 - Between providers, envs, branches
- Enable Flag-driven development
 - Review & track changes

How to use it?

```
1 name: Validate Flag Changes
2 on: pull_request
3
4 jobs:
5   validate-flags:
6     runs-on: ubuntu-latest
7     steps:
8       - uses: actions/checkout@v4
9         with:
10           fetch-depth: 0 # Required for git comparison
11
12       - name: Compare flag manifests
13         uses: open-feature/action@v1
14         with:
15           manifest: "flags.json"
16           # Automatically compares against PR base branch
```

open-feature/cli

+

open-feature/action



KubeCon

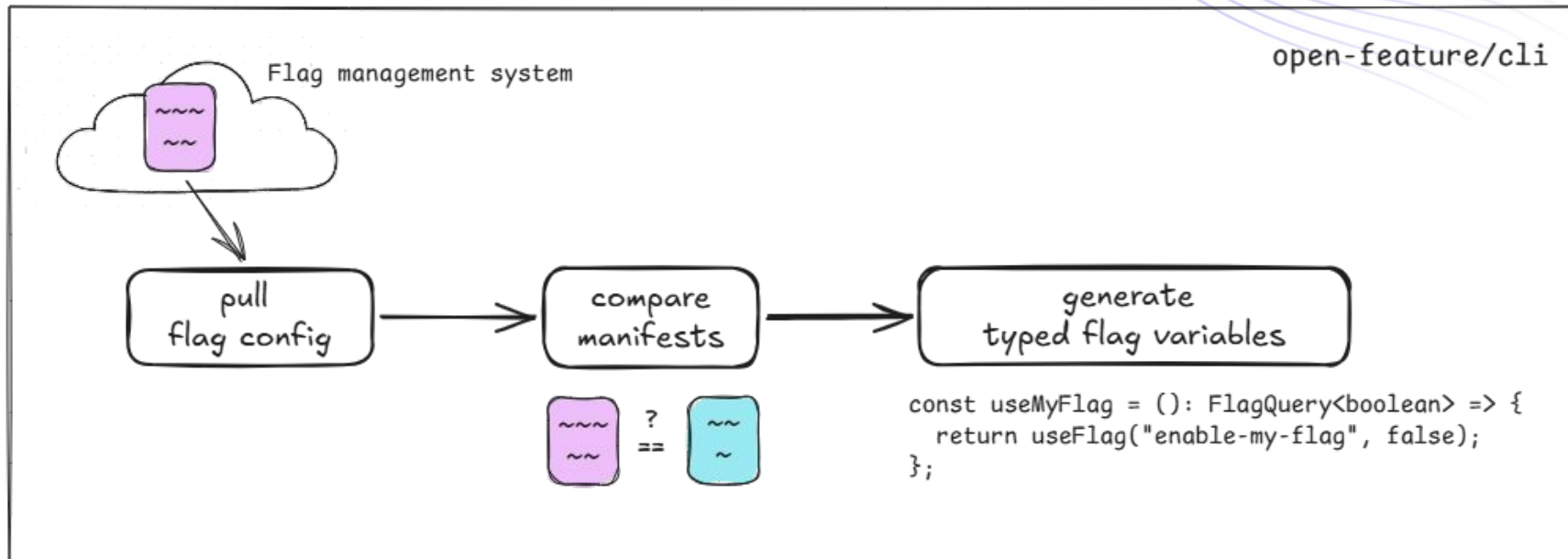


CloudNativeCon

Europe 2025

OpenFeature

How does it work?





Discussions & Decisions

Insights into an open-source community

API Singletons

1. The Singleton Problem

- **The Design:** Global singletons were an intentional choice to make integrations easy and ergonomic
- **The Friction:**
 - Micro-frontends with misaligned lifecycles
 - Dependency Injection frameworks (Spring, .NET) clashing with external singletons
 - Parallel tests trampling each other
- **The Goal:** Provide an opt-out for advanced use cases, without replacing the default singleton

API Singletons

2. The way towards a solution

- Called a dedicated session (Dec 18, 2025) outside regular meetings
 - **4 options discussed** on that day documented in [open-feature/spec#359](#)
- **PoC-first approach:** Multiple maintainers prototyped in parallel before writing spec
 - isolated API in JS SDK
 - provider-to-API locking mechanism
 - Go prototype

API Singletons

3. The solution

- **The Fix:** Introducing isolated OpenFeatureAPI instances
- **Safe by Design:** Uses a separate import path to prevent beginner mistakes
- **Consistent Experience:** Maintains the exact same API contract as the original singleton
- **Backwards Compatible:** The classic singleton remains the default; isolated instances are simply an escape hatch

Splitting API from Implementation

- **The Concept:** Separate the API contract (interfaces and types) from the underlying SDK implementation.
- **The Inspiration:** Adopting the proven OpenTelemetry (OTel) model.
- **Why it matters:**
 - **Stability:** The API remains stable while the implementation evolves freely
 - **Flexibility:** Providers target the stable API rather than getting locked to a specific SDK version
 - **Efficiency:** Test frameworks & providers only pull what they need
- **Status:** Currently under discussion as a roadmap proposal

OpenFeature – Feature Flagging Course



Introduction to Feature Flagging with OpenFeature

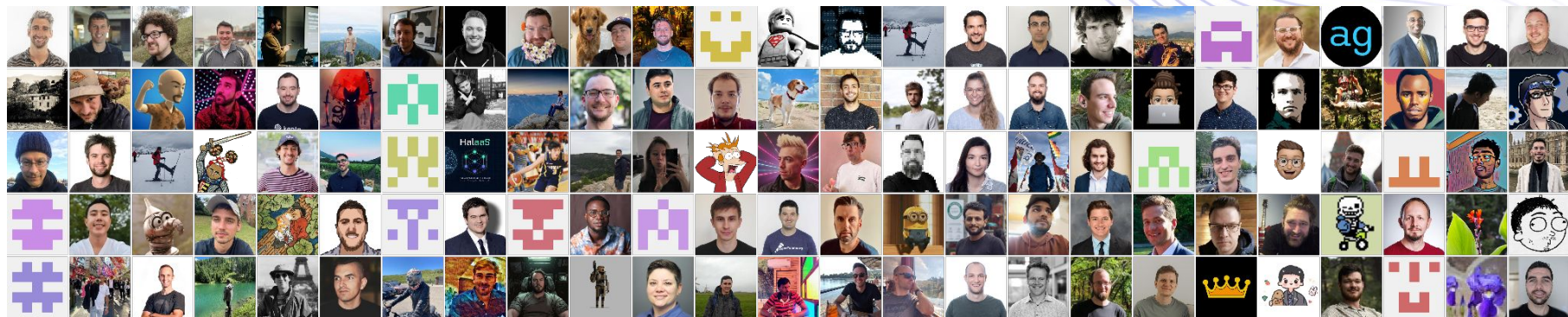
"The practical approach to feature flagging with OpenFeature was excellent. I appreciated learning how to **decouple deployment from release**, which is crucial for modern CI/CD pipelines. The vendor-neutral design of OpenFeature makes it easy to **integrate into existing infrastructure** without lock-in." (11/28/2025)

Key facts

- Online, Self Paced, \$0
- 10 Hours of Course Material
- Hands-on Labs and Quizzes, Discussion Forum



What's next?



 OpenFeature

Get started on:
openfeature.dev



Want to contribute?
[github.com/open-](https://github.com/open-feature)
[feature](https://github.com/open-feature)



Need help?
[#openfeature](https://openfeature.slack.com)
CNCF Slack



Standardizing Feature Flagging for Everyone

Learn more at openfeature.dev

