

Open Feature: Beyond Swapping Feature Flag Providers

Aurelijus Banelis

NTA Winter
2026-01-18



Aurelijus Banelis

Lead Software Engineer

aurelijus.banelis.lt

aurelijus@banelis.lt

PGP 0x320205E7**539B6203**
130D C446 1F1A 2E50 D6E3
3DA8 3202 05E7 539B 6203



Open Feature:

Beyond Swapping Feature Flag Providers

2 Open Feature:

3 Beyond Swapping

1 Feature Flag Providers

Feature flag

**Fast introduction via
visual DEMO**

OpenFeature

Integration code examples

Beyond

**OpenFeature extension points:
User privacy tiers
Dynamic test environments**

2 Open Feature:

3 Beyond Swapping

1 Feature Flag Providers



OpenFeature: Beyond Swapping Feature Flag Providers ☆

The feature flag ecosystem has matured into many competing platforms and SDKs—and now includes the **OpenFeature** specification.

The trap: feature flags quickly become provider-specific custom logic instead of reusable hooks/extensions.

This talk has three parts:

- Fast intro + visual demo: feature flags in practice
- OpenFeature SDK in reality: integration code examples and the trade-offs of using off-the-shelf tooling through OpenFeature
- Advanced use cases: OpenFeature ID extensions (for feature provider privacy, dynamic test environments)

After this talk, you should be able to:

- Reuse existing solutions by understanding the OpenFeature standard and the broader open source feature flag ecosystem
- Understand the trade-off between using advanced features from a single provider vs. composing capabilities via open standards
- Think more clearly about the real-world failure modes of feature flags

So, What Are Feature Toggles?

You hit deploy—and your checkout is down.

- In web systems, we ship code with features off by default and enable them via cookies, percentage rollouts, or lightweight A/B tests.
- In offline or embedded systems, the same idea appears as debug modes, experimental features, or configuration-driven behavior switches.

Different environments, same core idea: deploy now, decide later.

OpenFeature

Taking advantage of Standardized Feature flagging

FOSDEM

/dev/random

FOSDEM theme

T-SHIRTS AND HOODIES

T-shirts and Hoodies

	€25	€50
 T-shirt Limited edition		
 Hoodie Heavier knit		

DEMO

We have T-shirts and hoodies again! As each year, they are a special limited edition to show our gratitude for you donating to FOSDEM. However, we are having a virtual conference this time, so the T-shirt process is a bit different from last year. For example, you don't have to queue and they won't run out this year.

You can get **them till the end of the weekend after FOSDEM 2021** (2021-02-14). Don't miss out!

ORDER

Full DEMO example: <https://gist.github.com/aurelijusb/dccf6a276bd7e02571e55618c9c364b4>

T-SHIRTS AND HOODIES

T-shirts and Hoodies

	€25	€50
 T-shirt Limited edition		
 Hoodie Heavier knit		

We have T-shirts and hoodies again! As each year, they are a special limited edition to show our gratitude for you donating to FOSDEM. However, we are having a virtual conference this time, so the T-shirt process is a bit different from last year. For example, you don't have to queue and they won't run out this year.

You can get **them till the end of the weekend after FOSDEM 2021** (2021-02-14). Don't miss out!

ORDER

T-SHIRTS AND HOODIES

T-shirts and Hoodies

	€25	€50
 T-shirt Limited edition		
 Hoodie Heavier knit		

We have T-shirts and hoodies again! As each year, they are a special limited edition to show our gratitude for you donating to FOSDEM. However, we are having a virtual conference this time, so the T-shirt process is a bit different from last year. For example, you don't have to queue and they won't run out this year.

You can get **them till the end of the weekend after FOSDEM 2021** (2021-02-14). Don't miss out!

Features

OpenFeatureFosdemDEMO aurelijus@banelis.lt

Features

Add Feature

All Features Drafts

Search...

project owner tag type rules environment more

FEATURE KEY

TAGS

PRODUCTION

PREREQUISITES

DEFAULT

RULES

VERSION

LAST UPDATED

STALE



buy-button



FALSE

force

2

Jan 15, 2026



Test what values these features will return for your users from the Simulate page.

Features

OpenFeatureFosdemDEMO aurelijus@banelis.lt

Features

Add Feature

All Features Drafts

Search...

project owner tag type rules environment more

FEATURE KEY

TAGS

PRODUCTION

PREREQUISITES

DEFAULT

RULES

VERSION

LAST UPDATED

STALE



buy-button



FALSE

force

2

Jan 15, 2026



Test what values these features will return for your users from the Simulate page.

Full DEMO example: <https://gist.github.com/aurelijusb/dccf6a276bd7e02571e55618c9c364b4>

2 Open Feature:

3 Beyond Swapping

1 Feature Flag Providers

2 Open Feature:

3 Beyond Swapping

1 Feature Flag Providers

- ☐ Kotlin 8
- ☐ Ruby 7
- ☐ Rust 6
- ☐ Dart 2
- ☐ Next.js 1
- ☐ SvelteKit 1

Vendor

- ☐ ConfigCat 15
- ☐ DevCycle 14
- ☐ GO Feature Flag 14
- ☐ Kameleoon 10
- ☐ flagd 9
- ☐ Flagsmith 8
- ☐ LaunchDarkly 8
- ☐ Split 8
- ☐ Confidence by Spotify 7
- ☐ Flipt 7
- ☐ Hyphen AI 7
- ☐ GrowthBook 6
- ☐ VWO 6
- ☐ Bucketeer 5
- ☐ CloudBees 5
- ☐ OFREP 5
- ☐ Environment Variable 4
- ☐ FeatBit 4
- ☐ Multi-Provider 4
- ☐ AB Tasty 3
- ☐ Statsig 3
- ☐ Tggl 3
- ☐ Unleash 3
- ☐ AWS SSM 2
- ☐ Datadog 2
- ☐ Hypertune 2
- ☐ PostHog 2
- ☐ Reflag 2
- ☐ ConfigBee 1
- ☐ FFlags 1

React SDK

The official OpenFeature SDK for React

SDK

React JavaScript Client

Kotlin SDK

The official OpenFeature SDK for Kotlin targeting Android and other platforms.

Kotlin Client SDK

iOS SDK

The official OpenFeature SDK for iOS

Swift Client SDK

Ruby SDK

The official OpenFeature SDK for Ruby

Ruby Server SDK



Angular SDK

The official OpenFeature SDK for Angular

SDK

Angular JavaScript Client



Dart SDK

The official OpenFeature SDK for Dart

SDK

Dart Client



Rust SDK

The official OpenFeature SDK for Rust

SDK

Rust Client



Vercel Flags SDK

Vercel Flags SDK with OpenFeature Adapter

SDK

Next.js JavaScript Client



Vercel Flags SDK

Vercel Flags SDK with OpenFeature Adapter

SDK

SvelteKit JavaScript Client



AB Tasty JavaScript Node.js Provider

The official AB Tasty provider for OpenFeature

JavaScript Server Provider



AB Tasty JavaScript Web Provider

The official AB Tasty provider for OpenFeature

JavaScript Client Provider



AB Tasty Swift Client Provider

The official AB Tasty provider for OpenFeature

Swift Client Provider

Showing 1 to 20 of 205 results

Reuse

```
const featureName = 'feature1';  
if (growthbook.isOn(featureName)) {  
  console.log("feature is on");  
} else {  
  console.log("feature is off");  
}
```

Full example: <https://gist.github.com/aurelijusb/62d37ca32da7e30f42dfd132e319e1c7>



```
const featureName = 'feature1';  
if (growthbook.isOn(featureName)) {  
  console.log("feature is on");  
} else {  
  console.log("feature is off");  
}
```

```
// For A/B test
```

```
const details =  
growthbook.evalFeature(featureName);
```

```
{  
  "on": true,  
  "source": "experiment",  
  "experimentResult": {  
    "variationId": 1,  
    ...  
  }  
  ...  
}
```

0 Control	☐ FALSE
1 Variation	☒ TRUE

Full example: <https://gist.github.com/aurelijusb/62d37ca32da7e30f42dfd132e319e1c7>

```
const featureName = 'feature1';  
if (await client.getBooleanValue(featureName,  
false)) {  
  console.log("feature is on");  
} else {  
  console.log("feature is off");  
}
```

```
// For A/B test
```

```
const details = await  
client.getBooleanDetails(featureName, false);
```

```
{  
  "value": true,  
  "reason": "experiment",  
  "flagKey": "feature1",  
  "variant": "1",  
  "flagMetadata": {}  
}
```

0 Control	● FALSE
1 Variation	● TRUE



```
featureName := "feature1"
details := scoped.EvalFeature(ctx, featureName)
if details.On {
    log.Println("feature is on")
} else {
    log.Println("feature is off")
}
```

```
featureName := "feature1"
if client.Boolean(ctx, featureName, false, attrs)
{
    log.Println("feature is on")
} else {
    log.Println("feature is off")
}
details, err := client.BooleanValueDetails(ctx,
featureName, false, attrs)
```



Full examples:

<https://gist.github.com/aurelijusb/c1ba52d2e4f8c0bd537bd53e3776df26>

<https://gist.github.com/aurelijusb/e3d0a3222eb26b35c365a53afe2ab6a3>

feat: Growthbook experiment phase and seed #1

 Draft aurelijusb wants to merge 1 commit into `main` from `feat/growthbook-experiment-context` 

 Conversation 2  Commits 1  Checks 0  Files changed 1

Changes from all commits ▾ File filter ▾ Conversations ▾ Jump to ▾  ▾

▼  4  `libs/providers/growthbook/src/lib/translate-result.ts` 

```
@@ -24,10 +24,10 @@ export default function translateResult<T>(result: FeatureResult, defaultValues: T): T {
 24 24     val: result.value === null ? (defaultValues ? result.value : null) : result.value,
 25 25     reason: result.source,
 26 26     value: result.experimentRule?.key,
 27 +     flagMetadata: result.experiment?.phase && result.experiment?.seed ? {
 28 +       'experimentPhase': result.experiment.phase,
 29 +       'experimentSeed': result.experiment.seed,
 30 +     } : undefined
 27 31   };
 28 32
 29 33   if (FEATURE_RESULT_ERRORS.includes(result.source)) {
```

Limitations?

<https://github.com/aurelijusb/js-sdk-contrib/pull/1/files>

2 Open Feature:

3 Beyond Swapping

1 Feature Flag Providers

2 Open Feature:

3 Beyond Swapping

1 Feature Flag Providers

Introduction

Glossary

Types and Data Structures

Sections

Flag Evaluation API

Provider

Evaluation Context

Hooks

Events

Tracing

Appendix A: Included Suites

Appendix B: Gherkin Suites

Appendix C: OpenFeature Remote
Evaluation Protocol

Appendix D: Observability

Hooks add their logic at any or four specific stages of flag evaluation:

- `before`, immediately before flag evaluation



Web Consent Management Platform



We value your privacy

We and our partners are using technologies like Cookies or Targeting and process personal data like IP-address or browser information in order to personalize the advertisement you see. These technologies may access your device and help us to show you more relevant ads and save you time. We also use it in order to measure results for ads on our website content. Because we value your privacy, we ask you with asking your permission to use the following technologies. You can always change/withdraw your consent later by clicking on the settings button on the left lower corner of the page.

[Privacy Policy](#) [Legal Notice](#) [Privacy Settings](#)

Deny

Accept all and close

Powered by Usercentrics Consent Management

Hook: Application author/integrator-supplied logic that is called by the OpenFeature framework at a specific stage.

Stage: An explicit portion of the flag evaluation lifecycle. e.g. `before` being "before the `resolution` is run."

Invocation: A single call to evaluate a flag. `client.getBooleanValue(...)` is an invocation.

User privacy tiers

\$ npm run new

initial context: { id: '', consent: '' }

evaluated context: { id: 'a21481a5-5f02-4602-93d5-32ef37a0fa30', consent: '' }

Tracking { consent: 'new' }

feature is off

\$ npm run returning

initial context: { id: '00000000-1111-2222-3333-444444444444', consent: 'marketing' }

evaluated context: { id: '00000000-1111-2222-3333-444444444444', consent: 'marketing' }

Tracking { consent: 'marketing', id: '00000000-1111-2222-3333-444444444444' }

feature is off

\$ npm run withdraw

initial context: { id: '00000000-1111-2222-3333-444444444444', consent: 'withdraw' }

evaluated context: { id: '54d61cb1-b7ee-49aa-bbbe-6df408cd8d7e', consent: 'withdraw' }

Tracking { consent: 'withdraw' }

feature is off

\$ npm run developer

initial context: { id: 'developer-session', consent: 'marketing' }

evaluated context: { id: 'developer-session', consent: 'marketing' }

Tracking { consent: 'marketing', id: 'developer-session' }

feature is on

\$ npm run new

initial context: { id: '', consent: '' }

evaluated context: { id: 'a21481a5-5f02-4602-93d5-32ef37a0fa30', consent: '' }

Tracking { consent: 'new' }

feature is off

\$ npm run returning

initial context: { id: '000000000-1111-2222-3333-444444444444', consent: 'marketing' }

evaluated context: { id: '000000000-1111-2222-3333-444444444444', consent: 'marketing' }

Tracking { consent: 'marketing', id: '000000000-1111-2222-3333-444444444444' }

feature is off

\$ npm run withdraw

initial context: { id: '000000000-1111-2222-3333-444444444444', consent: 'withdraw' }

evaluated context: { id: '54d61cb1-b7ee-49aa-bbbe-6df408cd8d7e', consent: 'withdraw' }

Tracking { consent: 'withdraw' }

feature is off

\$ npm run developer

initial context: { id: 'developer-session', consent: 'marketing' }

evaluated context: { id: 'developer-session', consent: 'marketing' }

Tracking { consent: 'marketing', id: 'developer-session' }

feature is on

\$ npm run new

initial context: { id: '', consent: '' }

evaluated context: { id: 'a21481a5-5f02-4602-93d5-32ef37a0fa30', consent: '' }

Tracking { consent: 'new' }

feature is off

\$ npm run returning

initial context: { id: '000000000-1111-2222-3333-444444444444', consent: 'marketing' }

evaluated context: { id: '000000000-1111-2222-3333-444444444444', consent: 'marketing' }

Tracking { consent: 'marketing', id: '000000000-1111-2222-3333-444444444444' }

feature is off

\$ npm run withdraw

initial context: { id: '000000000-1111-2222-3333-444444444444', consent: 'withdraw' }

evaluated context: { id: '54d61cb1-b7ee-49aa-bbbe-6df408cd8d7e', consent: 'withdraw' }

Tracking { consent: 'withdraw' }

feature is off

\$ npm run developer

initial context: { id: '**developer-session**', consent: 'marketing' }

evaluated context: { id: '**developer-session**', consent: 'marketing' }

Tracking { consent: 'marketing', id: '**developer-session**' }

feature is on

```
import { Hook } from "@openfeature/server-sdk"; import { randomUUID } from "crypto";

type TrackViaCookies = { id?: string; consent: string; };

export const createPrivacyTierHook = (): Hook => ({
  before: (context) => {
    if (!context.context) {
      return;
    }
    if (context.context.consent !== "marketing") {
      context.context.id = randomUUID();
    }
  },
  finally: (context) => {
    const consent = String(context.context?.consent || "new");
    let track: TrackViaCookies = { consent };
    if (context.context?.consent === "marketing" && context.context?.id) {
      track = { ...track, id: String(context.context.id) };
    }
    console.log("Tracking", track);
  },
});
```

Full example: <https://gist.github.com/aurelijusb/c5a8b9aa8dc6d66448cbc76feaeaffcd>

Introduction

Glossary

Types and Data Structures

Sections

Flag Evaluation API

Provider

Evaluation Context

Hooks

Events

Tracking

Appendix A: Included Utilities

Appendix B: Gherkin Suite

Appendix C: OpenFeature Remote
Evaluation Protocol

Appendix D: Observability

Hooks add their logic at any or four specific stages of flag evaluation:

- `before`, immediately before flag evaluation
- `after`, immediately after successful flag evaluation
- `error`, immediately after an unsuccessful flag evaluation
- `finally`, unconditionally after flag evaluation

Dynamic test environments



Hooks can be configured to run globally (impacting all flag evaluations), per client, or per flag evaluation (locally). Some example use cases for a hook include adding additional data to the `evaluation context`, performing validation on the received flag value, providing data to telemetric tools, and logging errors.

Definitions

Hook: Application author/integrator-supplied logic that is called by the OpenFeature framework at a specific stage.

Stage: An explicit portion of the flag evaluation lifecycle. e.g. `before` being "before the `resolution` is run."

Invocation: A single call to evaluate a flag. `client.getBooleanValue(...)` is an invocation.

`http://localhost:4001/?session=`

```
{  
  "service": "service-a",  
  "cookie": "session=69f38d5f-772d-4a5b-af23-225aea4971da",  
  "downstreamPort": 4002,  
  "downstream": {  
    "service": "service-b",  
    "cookie": "session=69f38d5f-772d-4a5b-af23-225aea4971da",  
    "downstream": {  
      "service": "service-c",  
      "cookie": "session=69f38d5f-772d-4a5b-af23-225aea4971da",  
      "featureEnabled": false  
    }  
  }  
}
```

Full example: <https://gist.github.com/aurelijusb/19f8e8e2d4603a7f3453eae0395b702b>

```
http://localhost:4001/?session=developer-session
```

```
{  
  "service": "service-a",  
  "cookie": "session=developer-session",  
  "downstreamPort": 4002,  
  "downstream": {  
    "service": "service-b",  
    "cookie": "session=developer-session",  
    "downstream": {  
      "service": "service-c",  
      "cookie": "session=developer-session",  
      "featureEnabled": true  
    }  
  }  
}
```

Full example: <https://gist.github.com/aurelijusb/19f8e8e2d4603a7f3453eae0395b702b>

`http://localhost:4001/?session=test-environment-1`

```
{  
  "service": "service-a",  
  "cookie": "session=test-environment-1",  
  "downstreamPort": 4022,  
  "downstream": {  
    "service": "service-branch",  
    "cookie": "session=test-environment-1",  
    "downstream": {  
      "service": "service-c",  
      "cookie": "session=test-environment-1",  
      "featureEnabled": false  
    }  
  }  
}
```

Full example: <https://gist.github.com/aurelijusb/19f8e8e2d4603a7f3453eae0395b702b>

```
// 🧐 Feature Toggle Provider as a distributed state
const downstreamPort = await client.getNumberValue(
  "service-b",
  4002,
  featureContext,
);

// 😱 Feels dangerous, especially on Production:
const downstreamUrl =
`${downstreamHost}:${downstreamPort}`;
```

Full example: <https://gist.github.com/aurelijusb/19f8e8e2d4603a7f3453eae0395b702b>

```
// service-a.ts

const client = OpenFeature.getClient();
client.addHooks(secureTestEnvironmentsHook());

// test-env-hook.ts

import { Hook } from "@openfeature/server-sdk";

declare const process: { env: { ENVIRONMENT?: string } };

export const secureTestEnvironmentsHook = (): Hook => {
  finally: (context, evaluationDetails) => {
    if (context.flagKey !== "service-b") {
      return;
    }

    if (process.env.ENVIRONMENT !== "development") {
      evaluationDetails.value = context.defaultValue;
    }
  },
});
```

Full example: <https://gist.github.com/aurelijusb/19f8e8e2d4603a7f3453eae0395b702b>

```
npm run service-a-prod
```

```
{  
  "service": "service-a",  
  "cookie": "session=test-environment-1",  
  "downstreamPort": 4002,  
  "downstream": {  
    "service": "service-b",  
    "cookie": "session=test-environment-1",  
    "downstream": {  
      "service": "service-c",  
      "cookie": "session=test-environment-1",  
      "featureEnabled": false  
    }  
  }  
}
```

```
npm run service-a-dev
```

```
{  
  "service": "service-a",  
  "cookie": "session=test-environment-1",  
  "downstreamPort": 4022,  
  "downstream": {  
    "service": "service-branch",  
    "cookie": "session=test-environment-1",  
    "downstream": {  
      "service": "service-c",  
      "cookie": "session=test-environment-1",  
      "featureEnabled": false  
    }  
  }  
}
```

Full example: <https://gist.github.com/aurelijusb/19f8e8e2d4603a7f3453eae0395b702b>

2 Open Feature:

3 Beyond Swapping

1 Feature Flag Providers

Open Feature: Beyond Swapping Feature Flag Providers

Aurelijus Banelis

NTA Winter
2026-01-18