Engineer

Indicateur de fonctionnalité

active ou désactive le code sans reconstruire l'application

Ce que c’est

Un indicateur de fonctionnalité est un point de décision dans le code qui lit un réglage que l’équipe peut modifier sans reconstruire l’application. Derrière l’indicateur se trouve un nouveau chemin de code, déjà en production, mais qui reste inactif jusqu’à ce que le réglage l’active, pour tout le monde ou pour un groupe choisi d’utilisateurs. Pete Hodgson a décrit cette technique en détail sur le site de Martin Fowler, en plusieurs parties à partir de janvier 2016, puis avec une révision en octobre 2017. Il les appelle feature toggles et cite feature flags, feature bits et feature flippers comme autres noms du même mécanisme.

Hodgson décompose l’indicateur en plusieurs parties. Le point de bascule est l’endroit du code qui demande si le nouveau chemin est activé. Le routeur de bascule répond à cette question à partir de la configuration de bascule et, lorsque la réponse dépend de la personne qui pose la question, d’un contexte de bascule : généralement l’utilisateur courant, ou un cookie ou un en-tête HTTP spécial qui le représente. Il déconseille de coupler le point et le routeur. Au point de bascule, le code devrait poser une question précise, par exemple si cette confirmation de commande inclut le nouveau lien d’annulation. La règle qui en décide reste à un seul endroit, derrière le routeur, pour pouvoir changer sans modifier chaque point d’appel.

Les indicateurs diffèrent par leur durée de vie et la fréquence à laquelle leur réponse change. Hodgson les répartit en quatre catégories. Les indicateurs de mise à disposition permettent de déployer en production du code inachevé et non testé, sous une forme latente ; il indique qu’ils « ne devraient généralement pas subsister bien plus d’une semaine ou deux ». Les indicateurs d’expérimentation répartissent les utilisateurs en groupes pour un test A/B, chaque utilisateur restant toujours dans le même groupe. Les indicateurs d’exploitation permettent aux opérateurs de modifier le comportement du système en production. Beaucoup ont une courte durée de vie, mais quelques

interrupteurs d’arrêt durables permettent de réduire les fonctions dont le système peut se passer lorsque la charge est anormalement forte, par exemple un panneau de recommandations coûteux à produire. Hodgson les compare à un disjoncteur géré manuellement. Les indicateurs d’autorisation décident des fonctions qu’un utilisateur donné voit, par exemple les fonctions premium des clients payants ou les fonctions en version alpha réservées au personnel de l’entreprise. Ils peuvent rester des années.

Un indicateur de fonctionnalité n’est pas une mise en production canari, même si les deux vont souvent ensemble. Une mise en production canari fait fonctionner deux versions d’un service côte à côte et dirige une partie du trafic vers la nouvelle ; un indicateur déploie une seule version contenant les deux chemins et choisit entre eux à chaque requête. Une branche de longue durée dans le gestionnaire de versions sépare elle aussi le nouveau travail, mais le code n’arrive alors en production qu’à la fusion de la branche, tout d’un coup.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce qu'un indicateur de fonctionnalité ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Une seule équipe transforme une idée en un produit auquel vos clients peuvent se connecter, sans le faire passer entre trois prestataires.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. Application GIS : revenus multipliés par 10 — 10x croissance du chiffre d'affaires portée par le produit cartographique

Application GIS : revenus multipliés par 10 Contactez-nous →

Pourquoi c’est important

L’intérêt principal des indicateurs est de séparer le déploiement de la mise à disposition. Hodgson décrit les indicateurs de mise à disposition comme le moyen le plus courant d’appliquer un principe de la livraison continue : mettre une fonctionnalité à disposition est une étape distincte du déploiement de son code. Le code peut arriver en production chaque jour depuis une branche principale unique, tandis qu’une fonctionnalité développée sur un mois reste inactive jusqu’à être prête. C’est ce qui rend praticable le développement fondé sur une branche principale commune, ou trunk-based development.

Parce qu’un indicateur peut donner des réponses différentes selon l’utilisateur, une mise à disposition peut s’élargir par étapes. Hodgson décrit une activation initiale pour les utilisateurs internes, qu’il appelle Champagne Brunch, puisque

l'entreprise boit ainsi son propre champagne en premier. Vient ensuite une cohorte canari : son exemple choisit aléatoirement 1 pour cent des utilisateurs, à l'aide d'un modulo de l'identifiant utilisateur pour que les mêmes personnes restent dans ce groupe, tandis que les 99 pour cent restants conservent l'ancien comportement. Si cette cohorte signale des problèmes, l'indicateur est désactivé et personne n'a besoin d'annuler un déploiement.

Tout cela a un prix. Chaque indicateur ajoute une branche au code et une combinaison à tester, et sa configuration devient une partie de la production qui peut changer sans reconstruction. Hodgson recommande de conserver la configuration des bascules dans le gestionnaire de versions et de la modifier par des redéploiements chaque fois que l'indicateur le permet. La configuration suit ainsi le même pipeline que le code, et ses modifications sont examinées et enregistrées. C'est pour les indicateurs qui doivent changer en quelques minutes, tels ceux d'exploitation, que la configuration à l'exécution trouve toute son utilité.

À quoi cela ressemble en pratique

flagd est un moteur libre d'évaluation des indicateurs qui suit la spécification OpenFeature. Il n'a pas d'interface utilisateur et ne stocke rien en propre ; il lit les définitions dans un fichier JSON ou une autre source et répond aux demandes, en tant que sidecar, dans le processus de l'application ou comme service central. Voici un indicateur qui dirige environ un utilisateur sur vingt vers un nouveau parcours de paiement :

```
{
  "$schema": "https://flagd.dev/schema/v0/flags.json",
  "flags": {
    "new-checkout": {
      "state": "ENABLED",
      "variants": {
        "on": true,
        "off": false
      },
      "defaultVariant": "off",
      "targeting": {
        "fractional": [
          ["on", 5],
          ["off", 95]
        ]
      },
      "metadata": {
        "owner": "checkout",
```

```
        "expires": "2027-01-31"
      }
    }
  }
}
```

Chaque indicateur a un état et un ensemble de variantes qui doivent toutes être du même type ; la variante de repli est facultative. La règle `fractional` répartit les utilisateurs en groupes selon un poids relatif : 5 et 95 placent ainsi environ un utilisateur sur vingt dans `on` . Sans autre configuration, le groupe est calculé à partir de la clé de l'indicateur et du `targetingKey` dans le contexte d'évaluation. Le même utilisateur obtient donc toujours la même réponse, et renommer l'indicateur redistribue tout le monde. `metadata` contient ce que l'équipe souhaite joindre à l'indicateur, ici un responsable et la date à laquelle il devrait avoir disparu. Définir `state` à `DISABLED` fait répondre flagd sans aucune valeur, et l'appelant utilise alors sa propre valeur de repli.

Pour exécuter flagd avec ce fichier et lui demander une réponse par HTTP :

```
flagd start --uri file:./flags.json
curl -X POST "localhost:8013/flagd.evaluation.v1.Service/ResolveBoolean" \
  -H "Content-Type: application/json" \
  -d '{"flagKey":"new-checkout","context":{"targetingKey":"user-4711"}}'
```

La réponse contient la valeur, la variante et la raison `TARGETING_MATCH` . Dans l'application, l'appel passe par un client OpenFeature avec la clé de l'indicateur, une valeur de repli de `false` et le contexte d'évaluation. Le point de bascule ne voit qu'une valeur vraie ou fausse. Passer de 5 à 20 pour cent modifie le fichier des indicateurs, sans reconstruire l'application.

Les espaces de recrutement et de carrière des gens de mer, développés dans le cadre d'un poste occupé depuis 2025, montrent un indicateur de mise à disposition dans l'instantané d'implémentation consigné : le recrutement y est décrit comme en service et l'espace de carrière comme soumis à un indicateur de fonctionnalité, avec une mise à disposition incomplète. C'est un instantané historique, pas une déclaration sur l'état actuel du déploiement. Le back-office des opérateurs, également développé dans ce poste, est l'endroit où ces indicateurs sont modifiés. Les opérateurs les basculent et activent une

fonctionnalité pour certains utilisateurs seulement depuis un écran, au lieu qu'un ingénieur modifie à la main la base de production. Chaque modification est autorisée et enregistrée comme toute autre action dans cet outil.

Où les problèmes surviennent

Des indicateurs qui ne disparaissent jamais. Un indicateur de mise à disposition qui a rempli son rôle ne représente plus qu'un coût : une branche dont personne n'a besoin, un chemin que personne ne teste et un réglage que quelqu'un peut basculer par erreur. Hodgson propose de traiter les indicateurs comme un stock qui a un coût, et de limiter ce stock. Parmi ses remèdes figurent une tâche de suppression dans la liste des tâches à réaliser dès l'introduction d'un indicateur de mise à disposition, une date d'expiration, une bombe à retardement qui fait échouer un test ou empêche le démarrage de l'application une fois cette date dépassée, et une limite fixe du nombre d'indicateurs, de sorte qu'en ajouter un impose d'en retirer un autre. Retirer un indicateur signifie supprimer le chemin de code abandonné aussi bien que la vérification.

Un ancien indicateur reçoit un nouveau sens. Le 1er août 2012, le routeur automatisé d'ordres de Knight Capital, SMARS, a envoyé des millions d'ordres sur le marché en 45 minutes, et l'entreprise a perdu plus de 460 millions de dollars. La décision de la SEC explique comment. Le nouveau code destiné au Retail Liquidity Program du New York Stock Exchange réutilisait un indicateur porté par les ordres entrants, qui activait auparavant Power Peg. Knight avait cessé d'utiliser cette fonction en 2003, mais ne l'avait jamais supprimée. Le déploiement a omis un serveur sur huit ; sur ce serveur, l'indicateur réutilisé a donc réveillé l'ancienne fonction. Lorsque le personnel a ensuite retiré le nouveau code des sept autres, l'indicateur y a activé Power Peg également. Ne donnez jamais un nouveau sens à un indicateur retiré, supprimez le code mort avec l'indicateur qui le protégeait et vérifiez que tous les serveurs exécutent le même build avant d'activer un indicateur.

Personne n'a testé les combinaisons. Deux indicateurs donnent quatre combinaisons et dix en donnent plus de mille ; Hodgson parle d'explosion combinatoire. Il est rarement possible de toutes les tester. Testez donc celles qui peuvent atteindre la production : la configuration actuelle de production avec les indicateurs sur le point d'être mis à disposition activés, la solution de repli où ils sont désactivés et, comme le font aussi beaucoup d'équipes, tous les indicateurs activés. Une convention fixe aide : désactivé signifie toujours l'ancien comportement et activé le nouveau.

Personne n'a choisi le comportement de repli. La spécification OpenFeature impose que l'évaluation d'un indicateur ne lève pas d'exception. En cas de problème, l'appel renvoie la valeur de repli fournie par l'appelant, et la forme détaillée du résultat porte un code d'erreur. Le client ne devrait pas non plus écrire dans le journal, car une application déployée avant la définition de son indicateur l'inonderait sinon de `FLAG_NOT_FOUND`. L'application continue ainsi de fonctionner, mais une faute dans la clé d'un indicateur fait aussi appliquer silencieusement la valeur de repli à tous les utilisateurs. Fournissez l'ancien comportement sûr comme valeur de repli, puis ajoutez un hook, l'endroit où la spécification place la journalisation, pour compter les évaluations qui se terminent en erreur et déclencher une alerte lorsqu'elles apparaissent.

Tout le monde peut imposer une valeur. Il est pratique de permettre aux testeurs de forcer l'activation d'un indicateur avec un cookie spécial, un paramètre dans l'URL ou un en-tête HTTP. Hodgson souligne que les utilisateurs curieux ou malveillants peuvent imposer la même valeur et changer l'état d'un indicateur pour eux-mêmes. N'acceptez ces forçages que pour le trafic interne, ou signez-les cryptographiquement. Cela réduit le problème mais, comme le note Hodgson, ajoute sa propre complexité et sa propre surface d'attaque.

Pour aller plus loin

- Bascules de fonctionnalités, aussi appelées feature flags — l'article de Pete Hodgson sur le site de Martin Fowler : les parties d'une bascule, les quatre catégories, les tests et la limitation du stock.
- API d'évaluation des indicateurs — la spécification OpenFeature pour évaluer un indicateur, avec les valeurs de repli, les codes d'erreur et les raisons pour lesquelles l'évaluation ne lève jamais d'exception.
- flagd : définitions des indicateurs — le format JSON lu par flagd : état, variantes, variante de repli, ciblage et métadonnées.
- flagd : opération de répartition fractionnelle — les répartitions déterministes en pourcentage et ce qui décide du groupe dans lequel arrive un utilisateur.
- Affaire Knight Capital Americas LLC — la décision de la SEC de 2013 sur la perte boursière d'août 2012, avec l'indicateur réutilisé.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le processus de travail, les interfaces existantes et les utilisateurs qui ont besoin du changement. Nous pouvons définir une livraison testable et une transmission permettant à votre équipe d'assurer la maintenance des éléments livrés.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce qu'un indicateur de fonctionnalité ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/feature-flag/