Engineer

Відкат розгортання

повернення до попереднього розгортання після збою

Що це таке

Відкат розгортання повертає попередню встановлену версію або конфігурацію після зміни, що створила проблеми. Точна дія залежить від системи розгортання. Платформа може переспрямувати трафік на попередню версію, яка ще працює. Оркестратор може створити нові екземпляри застосунку зі збереженої ревізії. У Kubernetes відкат Deployment повертає шаблон Pod з попередньої ревізії. Залежні системи та база даних живуть за власним розкладом.

Скасування коміту в репозиторії створює нову ревізію вихідного коду. Її потрібно зібрати й розгорнути, щоб зміни з’явилися в робочій системі. Відкат може швидше використати пакет, який уже був зібраний. Після термінового втручання погодьте бажану версію в конфігурації з тим, що має працювати. Інакше звичайний процес може знову встановити версію, через яку стався збій.

Коли застосунок повертається до попередньої версії, у системі можуть залишитися нові записи, події та файли. Якщо попередня версія їх підтримує, не потрібно відкидати дії користувачів. Відновлення бази даних до певного моменту є окремою операцією. Для неї потрібно вибрати стан даних і спосіб перевірки. Наперед визначте, які частини системи слід відновити, а які відомості мають зберегтися.

Застосуйте поняття до рішення

Застосуйте «Що таке відкат розгортання?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Перевірено, випущено і з можливістю відкату, тож зміни доходять до клієнтів у вівторок удень, а не пізно в п'ятницю.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Автоматизувала розгортання GIS SaaS через CI/CD — розгортання, дані та звітність — усе без ручної роботи

Автоматизувала розгортання GIS SaaS через CI/CD Зв'язатися з нами →

Чому це важливо

Коли помилка починається після розгортання, команда має оцінити, що поверне корисну роботу служби. Доступний попередній пакет і відпрацьоване перемикання трафіку роблять відкат можливим варіантом. Іноді це швидше, ніж готувати термінове виправлення. Повернення версії може зупинити нові помилки, але зіпсовані записи чи вже виконані зовнішні дії потребуватимуть окремого розгляду. Зафіксуйте, яку саме частину збою має усунути втручання.

Запуск процесу дає тільки частину потрібних відомостей. Перевірте на контрольній операції, чи знову працює саме той сценарій, у якому виникла помилка. У службі замовлень це може бути тестова покупка. Підтвердьте версію, яка отримала запит, і перевірте повідомлення від черг та фонових завдань. Якщо помилка виникала лише під певним навантаженням, спостереження має його охопити.

Відкат потребує збереження пакетів та умов для їхнього запуску. Оркестратор може показувати ревізію, хоча образ уже вилучено з реєстру. Можуть спливати сертифікати або змінюватися права. Зберігайте потрібні артефакти на весь узгоджений період відновлення. Під час вправи користуйтеся тими правами та інструментами, які справді будуть доступні черговій особі.

Як це виглядає на практиці

Приклад повторюваного розгортання платформи в Azure описує визначений шлях через розробку, передвиробниче й робоче середовища. Відомий порядок допомагає планувати відновлення, бо команда може назвати версії та місця, де їх перевіряє. Приклад з портфоліо стосується розгортання. Наступна ситуація ілюструє, як окремо підготувати відкат.

Уявімо, що версія B служби замовлень дає збій, коли клієнт обирає збережену адресу. Версія A працювала перед зміною, а її пакет і налаштування доступні. Спершу перевірте, чи версія B не записала даних, яких A не розуміє. Якщо обидві версії підтримують поточний стан, поверніть

трафік до A й зробіть контрольне замовлення. Перевірте версію, яка його обробила, стан черги і відповідь сервера. Простежте також, що сталося із замовленням, розпочатим перед перемиканням. Якщо після переходу в черзі з’явилося повторне завдання, звірте його з уже створеним замовленням. Перевірка одного нового запиту не покаже, чи завершилася попередня робота. Запишіть, за якими ознаками команда помітила б втрату завдання або її наслідки.

Якщо B видалила колонку, яку використовує A, негайне повернення може додати нових збоїв. Можна тимчасово припинити запис у проблемній частині й запустити виправлення для наявної структури. Під час запланованої зміни спершу додайте нову структуру. Потім поступово переведіть ті частини системи, що читають і записують. Стару структуру видаляйте після того, як вона перестане бути потрібною сумісним версіям. Перевірте обидва напрямки переходу на проміжному етапі.

Де виникають проблеми

Успішний результат команди може означати, що система прийняла шаблон. Екземпляри ще можуть запускатися, а трафік може йти не туди. Після зміни перевірте версію, яка відповідає на контрольний запит, і покажіть, що виконано всі заздалегідь узгоджені критерії. Якщо первинна помилка повторюється, продовжіть пошук її причини.

Перемикач функції, правило трафіку або завдання за розкладом можуть діяти окремо від пакета застосунку. Після повернення старого образу інший процес усе ще може надсилати дані нового формату. Заздалегідь визначте потрібні налаштування, їхніх відповідальних і порядок змін. Передбачте, як поводяться під час переходу зовнішні клієнти та фонові служби.

Автоматичне розгортання може повернути проблемну версію, якщо вона досі записана в бажаному стані. Узгодьте зміну цієї конфігурації з поверненням пакета і трафіку. Запишіть ідентифікатори ревізій, час перемикання та відомі наслідки для користувачів. Коли виправлення буде готовим, розгорніть його звичним контрольованим шляхом і перевірте потрібні сценарії.

Що почитати далі

- Kubernetes про Deployments і відкат пояснює ревізії та роль шаблону Pod.
- Показники доставки DORA розрізняють швидкість доставки та відновлення після збою.
- PostgreSQL про зміну таблиць описує операції зі схемою.
- Helm rollback пояснює, як вибрати попередню ревізію.

Словник

Обирайте за фактичними умовами

Опишіть шлях розгортання, експлуатаційні ризики та чергову команду. Ми можемо визначити зміну з перевірками, шляхом відновлення та чіткою відповідальністю.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке відкат розгортання?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

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

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/deployment-rollback/