

Une approche de l'observabilité en trois étapes

De la réactivité à l'analytique data-driven, ou comment
améliorer l'expérience client

Sommaire

Introduction	03
Maturité de l'observabilité : 1. Réactivité	04
Maturité de l'observabilité : 2. Proactivité	07
Maturité de l'observabilité : 3. Analytique data-driven	12
L'observabilité dans New Relic	17

Introduction

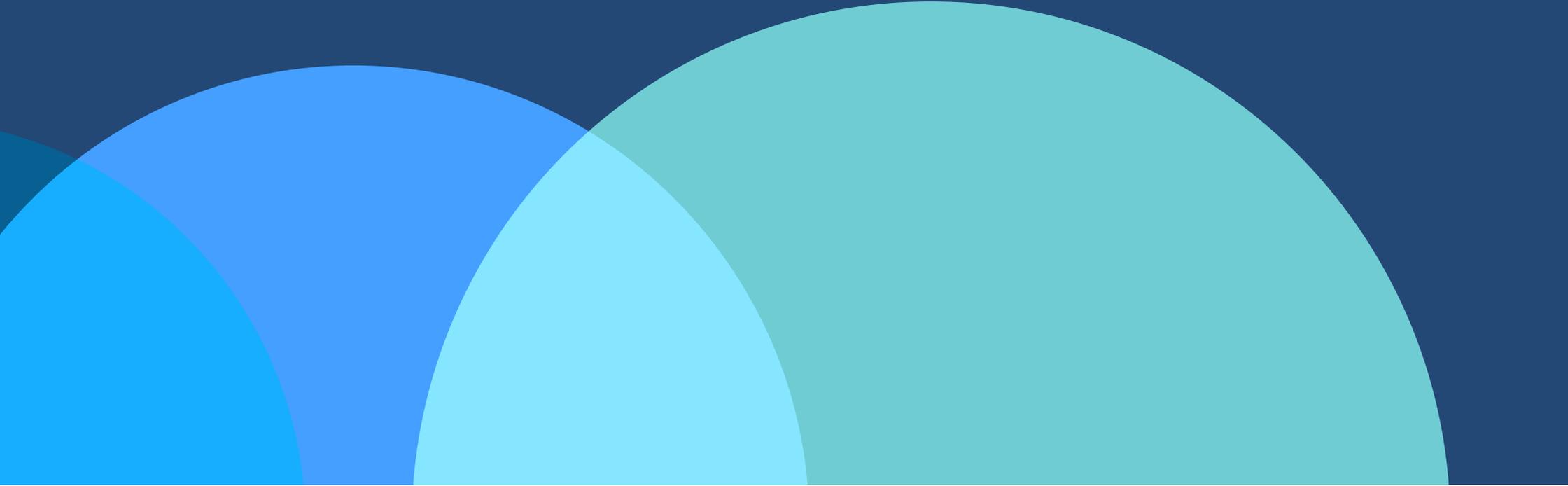
Que vous soyez dans la high-tech, la production, la santé, ou la finance, vos clients s'attendent à une expérience agréable à chaque interaction avec vous. Leur fidélité est, au mieux, fragile et pour y répondre, chaque entreprise essaie de déterminer comment passer à l'offensive avec ses applicatifs. Après tout, **les recherches montrent** que les clients à l'engagement élevé achètent 90 % plus souvent, dépensent 60 % de plus par achat, et apportent trois fois plus de valeur que les clients moyens. C'est pourquoi il vous faut une infrastructure qui puisse évoluer avec vos plus gros pics tout en continuant à livrer des expériences optimales qui font toute la différence.

Mais il y a un paradoxe : comment mettre en place ces expériences clients de pointe sans risquer de compromettre par inadvertance l'expérience client elle-même ? L'importance de cette dernière est bien documentée. **Une mauvaise expérience a plus de pouvoir qu'une bonne.** Nous avons tous **vu les gros titres** sur la perte de millions de CA en quelques heures et la disparition encore pire de la bonne volonté des consommateurs. Heureusement, il y a une bonne nouvelle : il n'est plus nécessaire que l'implémentation d'un nouveau code et l'utilisation d'une infrastructure moderne soient des propositions à haut risque. Si vous avez une **stratégie d'observabilité** en place, vous pouvez assurer une excellente expérience client avec vos logiciels malgré toute la complexité que présente l'entreprise numérique moderne.

New Relic définit les trois étapes de maturité de l'observabilité pour guider votre parcours : réactivité, proactivité, et analytique data-driven. L'objectif de toute stratégie d'observabilité est d'avoir un système automatisé qui s'autorépare et qui permet aux développeurs de comprendre en temps réel l'effet de ses mises à jour sur l'expérience numérique du client. Ce système doit aussi aider les ingénieurs à savoir non seulement qu'il y a eu une défaillance, mais également *pourquoi elle s'est produite*.

Cet ebook analyse en profondeur chaque étape de votre parcours d'observabilité et explique les activités et les KPI inhérentes à chacune.

Une bonne stratégie d'observabilité vous permet d'assurer une excellente expérience client avec vos applicatifs malgré toute la complexité que présente l'entreprise numérique moderne.



MATURITÉ DE L'OBSERVABILITÉ :

1. Réactivité

Nous avons tous entendu la phrase « il est impossible d'améliorer ce qu'on ne peut mesurer ». Par conséquent, une pratique robuste de l'observabilité commence par la mise en place de l'instrumentation pour rassembler autant de données télémétriques que possible. De là, les équipes peuvent prioriser l'élimination des problèmes flagrants.

Réactivité et recommandations

- Instrumentez et affinez
- Établissez des bases de référence et définissez les alertes fondamentales

KPI

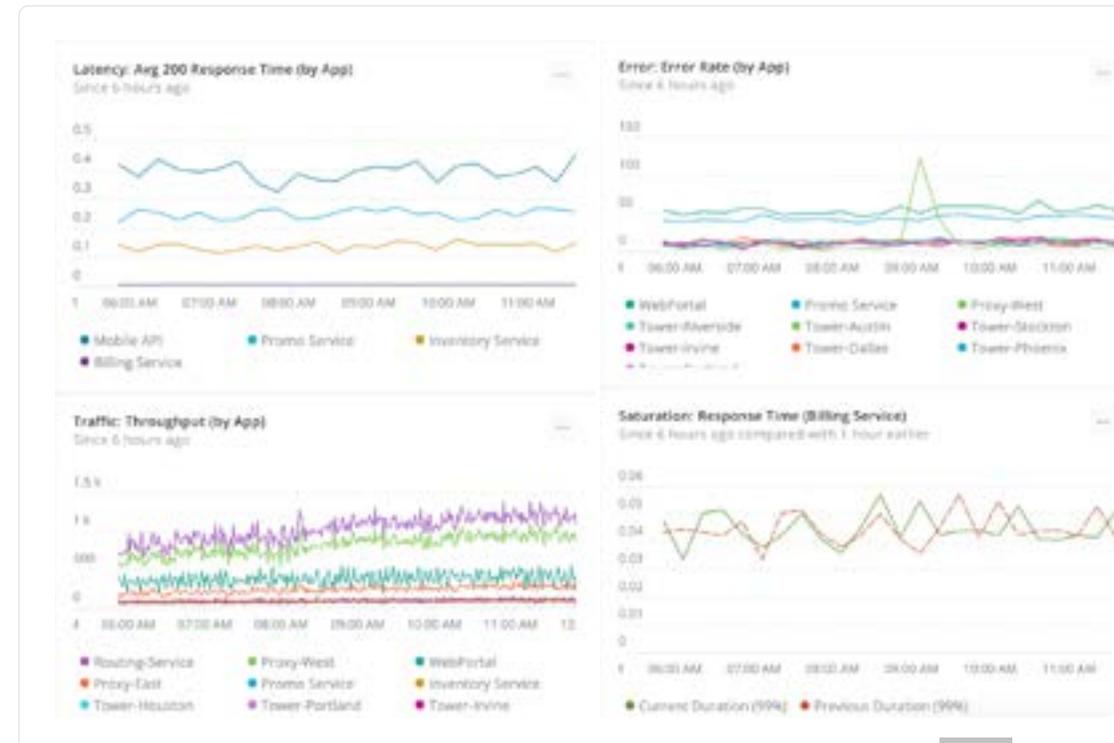
Erreurs, débit, latence, trafic, saturation, utilisation, Apdex

Instrumentez et affinez

Lorsque vous choisissez les éléments qui seront soumis au monitoring, impliquez les intervenants métier et les leaders du numérique de votre organisation. Votre objectif est de surveiller ce qui est important pour votre entreprise et de ne pas surcharger vos équipes avec trop d'alertes inutiles.

Les métriques, les événements, les logs et les traces (ou pour faire plus court, **MELT**) sont les types de données essentielles de l'observabilité. L'agrégation de données provenant de sources disparates, dont les **logs**, et le stockage dans un seul et même endroit, vous permet d'analyser, de visualiser et de dépanner vos systèmes à partir d'une plateforme unique. Cette capacité à instrumenter tout type de données, quelle qu'en soit la source (exclusive ou open source), est fondamentale à l'observabilité, car elle vous permet d'obtenir la télémétrie rapidement, quel que soit le type de données, et de résoudre les problèmes dynamiquement dès qu'ils surviennent.

En ayant toutes les données d'observabilité qui sont critiques à votre activité au même endroit, vous pouvez réduire la charge opérationnelle et les dépenses associées à la maintenance d'autres systèmes pour le stockage, les requêtes, l'affichage, et les alertes sur ces données.



Les quatre signaux dorés (tels qu'expliqués dans le manuel SRE de Google) — un point de départ pratique quand vous établissez les bases de référence de vos applications.

Cette capacité à instrumenter tout type de données quelle qu'en soit la source (exclusive ou open source) est fondamentale à l'observabilité.

Établissez les bases de références et définissez les alertes fondamentales

La collecte des métriques et des statistiques de performance sur vos applications et infrastructure vous permet d'identifier les goulots d'étranglement et les erreurs qui peuvent causer l'instabilité de vos applicatifs (et la frustration de vos utilisateurs). **Résolvez les problèmes sur vos applications** en vous attaquant aux transactions les plus lentes et aux erreurs les plus courantes. Vous pouvez ensuite établir une base de référence de ce qui est « normal » sur vos services et systèmes. Analysez les erreurs évidentes pour créer la **base de référence des conditions d'alerte** et ajustez la sensibilité selon vos besoins. Le mappage de la performance des transactions entre services jusqu'au produit est essentiel à l'élimination des silos entre produits, opérations, ingénierie, et équipes responsables du service.

MATURITÉ DE L'OBSERVABILITÉ :

2. Proactivité

Avec autant de signaux provenant de différents outils, il est difficile de déterminer et d'évaluer la cause profonde des incidents rapidement, et encore plus de détecter et de résoudre les problèmes de manière proactive. La complexité accrue en termes de technologie et de processus rend difficiles l'analyse et l'interprétation de milliers « d'inconnues inconnues ». Ce stade permet d'harmoniser les équipes en définissant les SLO (objectifs de niveau de service) et en optimisant vos processus afin d'équilibrer fiabilité du système et rapidité, mais aussi de résoudre les problèmes avant qu'ils n'impactent vos clients.

Proactivité et recommandations

- Définissez les SLO
- Optimisez la stratégie d'alerte
- Réduisez le temps de réponse aux incidents
- Activez des boucles de rétroaction rapides

KPI

Disponibilité, erreurs, latence, trafic, saturation, utilisation, fréquence de déploiement, délais d'exécution des modifications, temps moyen de détection (MTTD), temps moyen de résolution (MTTR), temps moyen entre pannes (MTBF), temps de chargement des pages

Définissez les SLO pour harmoniser les équipes

Les **SLO** (objectifs de niveau de service) établissent un mécanisme efficace qui permet de coder les objectifs de l'équipe de façon à ce qu'ils puissent être mesurés et partagés. Ils placent également des limites claires au niveau des perspectives de service, ce qui aide les équipes à être plus rapides et à avoir plus de liberté pour expérimenter avec de nouvelles approches. Si vous n'êtes pas certain de savoir où commencer, les **quatre signaux dorés** (latence, trafic, erreurs et saturation) présentés dans le **manuel SRE de Google** peuvent vous aider à faire la lumière sur les problèmes. Trouvez ensuite un ou deux SLI (indicateurs de niveau de service) de haut niveau représentant l'état de santé général de vos services. Par exemple, les services de streaming vidéo se servent du délai d'affichage de la première image, c'est-à-dire le délai entre la sélection d'un film par l'utilisateur et l'apparition de la première image.

Optimisez la stratégie d'alerte

Une fois l'agrégation des données en place et les objectifs de l'équipe clairement définis, vous aurez besoin d'une méthode systématique et fiable pour mobiliser les équipes d'astreinte dès qu'un problème survient. Des alertes bien définies vous aident à comprendre l'état de santé de vos systèmes et vous permettent de répondre rapidement aux problèmes de performance.

Les SLO n'ont pas tous besoin de se transformer en alerte. Une bonne stratégie consiste à prendre les SLO et à créer un groupe d'alertes simples et exploitables. De nombreuses organisations matures définissent généralement moins d'alertes et les concentrent sur un ensemble clé de métriques qui indiquent quand il y a une réelle détérioration de l'expérience client. Les équipes DevOps utilisent souvent **Apdex** dans le cadre de leur stratégie d'alerte afin d'aligner les alertes avec les indices de baisse de la satisfaction des utilisateurs.

Les SLO n'ont pas tous besoin de se transformer en alerte. Une bonne stratégie consiste à prendre les SLO et à créer un groupe d'alertes simples et exploitables.

Éléments essentiels d'une bonne stratégie d'alerte

Lorsque vous élaborez votre stratégie d'alerte, posez-vous cette question : « si le client n'est pas impacté, est-ce vraiment nécessaire de réveiller quelqu'un ? »

Question	Métriques et KPI
Sommes-nous en activité ?	Définissez des pings et des alertes en fonction de la disponibilité.
Quel est l'état de notre infrastructure sous-jacente ?	Gérez et dépannez vos hôtes et conteneurs.
Quel est l'état de santé de notre application ?	Utilisez des métriques d'utilisateur réelles pour comprendre le backend. Utilisez des métriques et tracez les données avec des outils open source, puis affichez ces informations aux côtés de toutes les autres données systèmes et services que vous gérez.
Comment puis-je dépanner une erreur système ?	Faites des recherches et analysez la cause profonde d'un problème sur tous les logs concernant l'application et l'infrastructure.
Quelle est la qualité globale de notre application ?	Utilisez un score Apdex pour rapidement évaluer la qualité d'une application.
Comment se débrouillent nos utilisateurs ?	Faites le monitoring de l'expérience utilisateur en frontend et sur mobile.
Comment va l'entreprise ?	Concentrez-vous sur les transactions clés d'une application et associez-les aux résultats prévus de l'entreprise afin de corréliser les performances de l'application avec celles de l'entreprise.

Réduisez le temps de réponse aux incidents

Chaque minute passée à interpréter manuellement vos données télémétriques pour diagnostiquer et résoudre les problèmes a un impact sur la réputation et les résultats de votre entreprise. C'est pourquoi vous avez besoin de fournir à vos équipes IT opérationnelles l'information et l'automatisation qui leur donne les moyens de détecter et résoudre les incidents plus rapidement et de manière proactive.

Attribuez la responsabilité de la santé de l'application et des services à une personne ou équipe. Elle servira de premier intervenant en cas d'incident. Assurez-vous que les communications pendant les incidents critiques se déroulent sur des canaux facilement accessibles et très visibles. Par exemple, si vos équipes communiquent habituellement sur un outil tel que Slack, faites en sorte que les alertes en cas d'incidents puissent être envoyées vers Slack.



Servez-vous des marqueurs de déploiement pour comprendre rapidement l'impact sur la fiabilité que peuvent avoir les modifications apportées au code.

Activez des boucles de rétroaction rapides

Les boucles de rétroaction rapides sur les nouvelles versions réduisent le MTTR et les risques de perte de robustesse associés aux mises à jour fréquentes. Il est important de faire le suivi des déploiements et de l'impact sur l'expérience client que peuvent avoir les modifications apportées au code et à l'infrastructure.

Le suivi des déploiements est très utile pour déterminer la cause profonde des détériorations immédiates, à long terme, ou progressives dans votre application. Les boucles de rétroaction rapides sont également essentielles lors des déploiements modernes de type canary ou **blue/green**, par exemple.

Les boucles de rétroaction rapides sur les nouvelles versions réduisent le MTTR et les risques de perte de robustesse associés aux mises à jour fréquentes.

MATURITÉ DE L'OBSERVABILITÉ :

3. Analytique data-driven

L'objectif de l'observabilité est fondamentalement de permettre aux équipes d'agir au niveau des données partagées, de connecter les personnes aux processus et aux performances IT dans toute l'organisation et de relier le tout à des résultats commerciaux précis. Pour avoir un flux continu des données d'observabilité, automatisez l'instrumentation pour les nouveaux services et applications. Vous aurez ensuite la possibilité d'agir rapidement en toute confiance, car vous serez une organisation orientée sur les données grâce à l'analytique data-driven.

Analytique data-driven et recommandations

- Analysez les tendances SLO
- Visualisez les relations entre développement, opérations et client
- Automatisez l'instrumentation

KPI

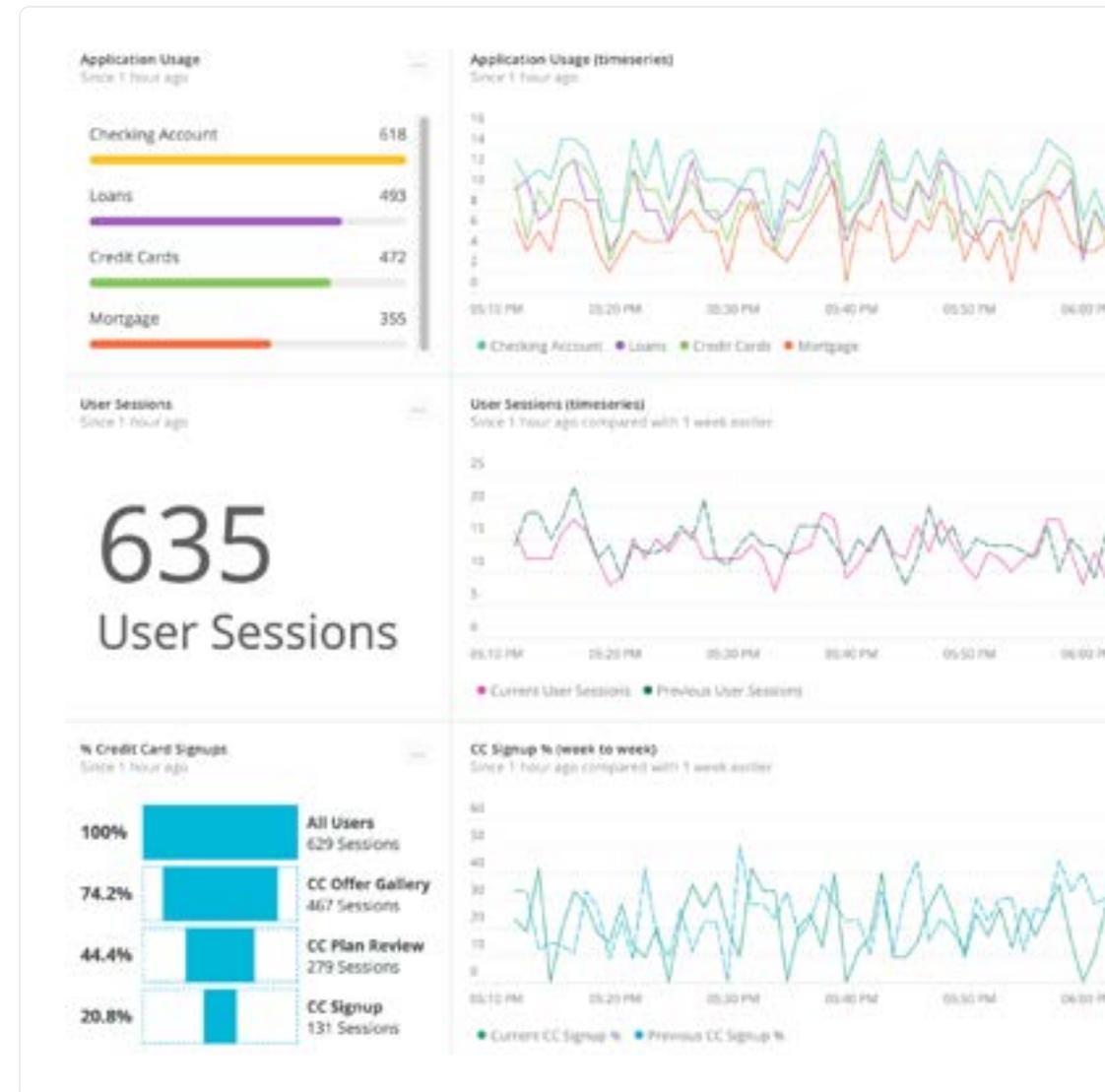
Nombre et MTTR des incidents dus à des modifications, satisfaction du développeur, satisfaction du client, conformité aux SLO et SLA, utilisation, taux d'erreur, alertes actives, nombre de tickets d'assistance

Analysez les tendances SLO

Le fait de définir et de mesurer les SLO vous permet de décomposer votre applicatif en un ensemble d'indicateurs clés auxquels vous pouvez avoir recours pour valider le comportement prévu des services et faire le suivi de l'impact des incidents sur les prestations. Servez-vous du travail de définition des SLO que vous avez fait (au cours de l'étape 2. Proactivité) pour trouver les points sensibles sur lesquels il faut se concentrer. Les dashboards d'équipe sont un bon point de référence pour décrire l'accomplissement des SLO au fil du temps.

Visualisez les relations entre développement, opérations et client

Un dashboard organisé autour de la performance métier donnera à vos équipes un aperçu de l'expérience des utilisateurs avec votre application. Si vous savez comment vos clients utilisent (ou abandonnent) vos applications, vous pouvez prioriser vos tâches futures. Avec une bonne culture d'observabilité vos données ne sont plus seulement accessibles par les seuls utilisateurs du backend, mais elles sont aussi mises à la disposition de différents groupes, comme le service client, l'assistance, l'équipe produit, la vente, le marketing et les cadres dirigeants. Quand elles sont placées sur le devant de la scène, elles s'avèrent très motivantes pour les équipes.



Établissez les liens vous permettant de comprendre l'impact sur les résultats métier que peuvent avoir les performances de l'application et de l'infrastructure.

Automatisez l'instrumentation

L'automatisation peut vous rendre plus efficace, elle peut aussi aider à éliminer les erreurs humaines, et ce qui est encore plus important, améliorer l'observabilité. Vous passerez moins de temps à instrumenter les systèmes et vous réduirez les charges de travail au fur et à mesure que les stacks de l'application et de l'infrastructure augmentent inexorablement. Pour inculquer cette discipline, pensez à mettre une limite à 40 % sur le travail opérationnel pour tous les SRE, tels que les tickets, les astreintes, et les tâches manuelles. Une équipe SRE doit passer le reste du temps à développer les applications. Après un petit moment, votre équipe SRE devrait avoir une charge opérationnelle très faible et être occupée presque entièrement aux tâches de développement, et le service devrait tourner et se dépanner tout seul. Optimisez les stratégies d'observabilité qui permettent à vos équipes de minimiser les tâches et d'optimiser l'automatisation.

Les 10

meilleures pratiques de l'observabilité

1

Collectez des données

Rassemblez le plus de données de télémétrie possible sur vos systèmes (métriques, événements, logs et traces, en open source ou propriétaires) et stockez-les sur une plateforme unique.

OBJECTIF

Collecter et explorer toutes vos données de télémétrie sur une plateforme uniformisée.

2

Établissez des bases de référence

Analysez les erreurs flagrantes pour créer des conditions d'alertes de base et ajustez la sensibilité en fonction de vos besoins.

OBJECTIF

Éliminer les problèmes flagrants, puis développer une base de référence qui priorise l'action.

3

Créez des SLO

Définissez les SLO, en commençant peut-être avec les quatre signaux dorés (latence, trafic, erreurs et saturation) et quelques métriques pertinentes qui mesurent l'expérience de l'utilisateur final.

OBJECTIF

Coder les objectifs de l'équipe de sorte qu'ils puissent être mesurés.

4

Définissez les alertes

Définissez une stratégie d'alerte simple qui priorise un ensemble clé de métriques indiquant que l'expérience client s'est détériorée.

OBJECTIF

Se concentrer sur les alertes pour les SLO les plus critiques.

5

Réduisez les temps de réponse aux incidents

Fournissez à vos équipes IT opérationnelles l'information et l'automatisation qui leur donnent les moyens de détecter et résoudre les incidents plus rapidement et de manière proactive.

OBJECTIF

Optimiser la boucle de rétroaction entre la détection des problèmes et leur résolution.

6

Faites le suivi des déploiements

Faites le monitoring des déploiements d'applications afin de pouvoir évaluer l'impact sur la performance des modifications apportées au code.

OBJECTIF

Déterminer la cause profonde des détériorations immédiates, à long terme, ou progressives dans votre application.

7

Automatisez

Maintenez un flux continu des données télémétriques en automatisant l'instrumentation pour les nouveaux services et applications.

OBJECTIF

Permettre à vos équipes d'agir rapidement en toute confiance avec des données télémétriques partagées.

8

Analysez

Créez des dashboards pour mesurer l'accomplissement des SLO au fil du temps.

OBJECTIF

Valider les attentes en termes de livraison de service et trouver les points sensibles sur lesquels il faut se concentrer.

9

Soyez à l'écoute du client

Donnez à vos équipes un aperçu sur la façon dont les clients utilisent (ou abandonnent) vos applications.

OBJECTIF

Améliorer la façon dont vous triez et priorisez la charge de travail future.

10

Minimisez la charge de travail

Inculquez la discipline de l'automatisation en limitant le travail de maintenance de l'équipe SRE à 40 %.

OBJECTIF

Avoir très peu de charge opérationnelle pour que les équipes SRE interviennent plutôt dans des tâches de développement et avoir un service qui tourne essentiellement tout seul et se répare lui-même.

L'observabilité sur New Relic

Maintenant que vous en savez plus sur les trois étapes de la maturité de l'observabilité, voici quelques conseils pratiques pour bien démarrer avec New Relic.

Réactivité

- Installez les agents de monitoring adéquats pour les applications, l'infrastructure sous-jacente, et l'expérience utilisateur en frontend.
- Collectez les données auprès de **services cloud** populaires, tels qu'Amazon Web Services (AWS), Microsoft Azure et Google Cloud Platform, mais aussi les **données télémétriques critiques de n'importe quelle source**, comme OpenTelemetry, Envoy, Dropwizard, Prometheus, etc.
- Accédez aux **intégrations sur hôte** par le biais de l'agent Infrastructure.
- Établissez des **transactions clés** dans New Relic APM qui vous donneront un niveau personnalisé de monitoring, ou des **règles d'alertes uniques**.

Proactivité

- **Configurez New Relic Synthetics** pour interagir avec l'intégralité du système comme le ferait un utilisateur externe, afin que vos équipes puissent réaliser des vérifications de haut niveau sur la performance, la disponibilité et l'expérience utilisateur.
- Implementez **New Relic AI** pour uniformiser et améliorer les incidents de production.
- Utilisez la fonctionnalité de marqueurs de déploiement de **New Relic APM** pour **enregistrer les déploiements** de chaque application. **Les marqueurs de déploiement** dans les dashboards de l'équipe vous permettent d'analyser l'impact sur la performance avant et après une sortie de logiciel.

Analytique data-driven

- Profitez de l'**application SLO/R** dans New Relic One pour définir les SLI et établir les cibles des SLO. Envisagez de communiquer l'accomplissement des SLO. Vos équipes devraient comparer leur propre évolution aux SLO cibles et s'en servir d'indicateur pour savoir quand il faut ralentir ou quand il faut pousser de nouvelles fonctionnalités. Vous pouvez également utiliser les tendances SLO pour faciliter la prise de décision organisationnelle.
- Utilisez ce **tutoriel** pour en savoir plus sur l'expérience des clients avec vos applications **en les regroupant** par région, type d'appareil, etc. Segmentez et évaluez les groupes d'utilisateurs finaux et assurez-vous que vous optimisez cette expérience conformément à la stratégie métier globale.
- Utilisez les **dashboards de New Relic One** pour obtenir des vues modulables et interactives de données, où qu'elles se trouvent sur la plateforme.
- Utilisez l'**API Synthetics REST** pour créer et gérer les moniteurs New Relic Synthetics.
- Configurez les paramètres d'alerte et **définissez les conditions d'alerte**.
- **Créez un script ou un plug-in de compilation** qui capture les dépendances et les tâches pour terminer l'installation de l'agent et fournir des configurations par défaut qui soient pertinentes.

Vous êtes prêt à vous lancer dans l'observabilité ?

Inscrivez-vous pour obtenir votre **compte New Relic gratuit**.

