

2020年8月25日

株式会社インプレスR&D

<https://nextpublishing.jp/>

Amazon Web Service (AWS) の監視設計を学ぶ！

『AWS を使って学ぶ監視設計』発行

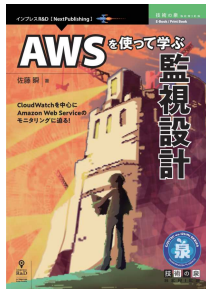
技術の泉シリーズ、8月の新刊

インプレスグループで電子出版事業を手がける株式会社インプレス R&D は、『AWS を使って学ぶ監視設計』(著者:佐藤 瞬)を発行いたします。

最新の知見を発信する『技術の泉シリーズ』は、「技術書典」や「技術書同人誌博覧会」をはじめとした各種即売会や、勉強会・LT 会などで頒布された技術同人誌を底本とした商業書籍を刊行し、技術同人誌の普及と発展に貢献することを目指します。

『AWSを使って学ぶ監視設計』

<https://nextpublishing.jp/isbn/9784844378747>



著者:佐藤 瞬

小売希望価格:電子書籍版 1600 円(税別)／印刷書籍版 2000 円(税別)

電子書籍版フォーマット:EPUB3／Kindle Format8

印刷書籍版仕様:B5 判／カラー／本文 70 ページ

ISBN:978-4-8443-7874-7

発行:インプレス R&D

<<発行主旨・内容紹介>>

この本では、「モニタリング(監視)」というテーマについて、CloudWatchを中心にAWS(Amazon Web Service)の各サービスを見ながら、どのように監視を設計するかを著者の経験を基に説明します。

「監視設計」は、システム開発の最初から携わっていれば経験することができますが、運用から参加する場合は既存の仕組みを使うことが多くなります。本書では実際に設計するときに役立つ知識を丁寧に解説します。

(本書は、次世代出版メソッド「NextPublishing」を使用し、出版されています。)

監視の設計を基本から解説

テムであれば、サーバーが稼働しているかが重要になり、オートスケールするサーバーであれば、無用なコストをかけていないかも重要になります。そのため、監視設計を考える際には、自分たちの監視対象はどういうものなのかを理解することが、最初のステップになります。

自分たちは何を確認すべきか明確にする

次に、監視対象を理解した上で「自分たちは何を確認すべきなのか」を明確にします。単純なことではありますが、思った以上に努力を使う作業です。というのも、考えていくうちに、全て重要に見えてしまうことが多いからです。しかし、あれもこれもとなると、いざ運用がはじまったときに、アラートに振り回されることになります。特に現在は、データストアひとつとってもRDB以外に多くの選択肢があり、マイクロサービスの流れによりアプリケーションが細分化されています。これによりコンポーネントが増え、同時にメトリクス数も膨大な数になっています。不安になる心を律して、本当に重要な項目を明確にします。

ユーザー視点で考える

「自分たちは何を確認すべきなのか」を考える際に重要なのは、ユーザー視点で考えることです。たとえば、「I/Oが増加すればCPU使用率が上がり、アプリケーションがハングして、サービスがダウンする可能性がある。だからI/Oを確認しよう」というように、障害に対する原因となり得るメトリクスを洗い出して監視項目を設定していくと、やりがありません。そのように考えていくと、確実に運用が破綻します。サービスを利用する目的はユーザーに価値を届けることです。なので、あくまでユーザーに届ける価値が損なわれていない、または低下していないことが重要です。CPU使用率が100%だったり、メモリが足りなくなったりしたとしても、ユーザーに価値を提供できていればそれは問題はありません。

ユーザーに近いメトリクスを重要視する

CPU使用率やメモリなどのコンピューターリソースに関するメトリクスは、ユーザーからは非常に「近い」メトリクスで、直接的には関係がありません。逆に、HTTPリクエストのエラーやレスポンスタイムなどは、ユーザーに直接的に關係する非常に「近い」メトリクスといえます。こういったメトリクスは、サービスの価値に直結することも多いので、このようなメトリクスをどのように収集、または表現するかを考えることが重要です。

メトリクスは可能な限り多く収集しておく

重要視するメトリクスはユーザーに「近い」ものですが、サーバー関連のメトリクスは不要という話ではありません。たとえば、エラー率が増加した際に何か問題が起こっていることはわかりますが、原因はそれだけではわかりません。原因の調査時（ドリルダウン）には詳細なメトリクスが必要になります。何が原因になるかはわかりませんが、取れるメトリクスはなるべく収集して、保管してしまえばいいと考えています。ストレージが限られている時代なら慎重になりますが、現在はAWSを始めとする各クラウドが充実していますので、ストレージに困ることもありません。

もちろんコストが許す限りにはなりますが、メトリクスのデータもそこまでの容量にはなりませんので、基本的には収集する方針で設計します。

アラートは最低限に

メトリクスはなるべく多く収集しますが、アラートを設定するのは最低限に留めます。多く設定するとアラートに振り回されて運用になりません。また、多くのアラートが来ると形骸化して気に留めなくなり、かえって発生している問題を見逃すこととなります。しかし、長くシステムを運用すれば、アラートを増やすこともあります。人間の心理の問題で、アラートを増やすのは簡単ですが、外すのは非常に難しいです。「外してしまっ、もし何か起きたら……」と考えると、なかなか外せません。担当者レベルでは問題なくとも、リーダーやマネージャーから見れば、問題ないと確信が持てないこともあるでしょう。そのため、本当に重要なメトリクスに絞り、アラートは最低限にすることを意識します。

ビジネス視点のメトリクス

システム運用という観点では、ユーザー視点で見てメトリクスを収集し、アラートを設定すればよいです。ですが、サービスを継続するためには、ビジネス視点でもサービスを監視する必要があります。ビジネス視点の監視項目としては、アクセス数やアクティブユーザー数、滞在時間など、ECサイトであれば売上や商品数などがあるでしょう。これらをも最も使うのは、サーバー開発・運用をするチームとは異なると思いますが、設計としては、そういったメトリクスも収集できるようにする必要があります。また、サービスのビジネス状況を把握することはエンジニアにとっても重要なので、設計段階で考慮しましょう。

1.3 監視に必要な処理

監視を行うために必要な処理は以下になります。

1. メトリクスの収集
2. メトリクスの保管
3. メトリクスの抽出・集計
4. メトリクスの可視化
5. アラート管理・通知

まずはメトリクスを収集し、保管します。保管したメトリクスを条件に応じて抽出し、合計や平均、パーセンタイルなどを集計します。抽出・集計した結果は、折れ線グラフや棒グラフなど、人が見やすいように可視化します。可視化した結果（ダッシュボード）を24時間監視できればいいですが、人間には難しいです。なので条件を設定し、条件を満たした場合に通知してもらうよう設定します。これが監視の流れになります。

設計のポイント

それでは、それぞれの処理において、監視を設計する際に押さえておくポイントを見ていきます

AWS における監視設計を解説

第3章 AWSにおける監視設計

この章では、AWSの監視サービスを項目ごとに紹介します。各サービスの詳細までは触れませんが、AWS全体として提供している監視機能を把握いただければと思います。

3.1 AWSにおける監視設計の方針

AWSの監視に利用できるサービスは、主にCloudWatch、X-Ray、Elasticsearch Serviceです。CloudWatchはメトリクスの検索、保管、可視化、計算ができます。X-RayはAPM（Application Performance Management）サービスです。Elasticsearch Serviceはその名の通り、Elasticsearchのマネージドサービスです。

「監視」という業務は多岐にわたります。CPU使用率を見るのも監視ですし、アクティブユーザー数を見るのも監視です。これらの業務をすべてAWS上だけで完結しようとする、どこかで断絶が出てきます。もちろん、CloudWatchはカスタムメトリクスとして任意のメトリクスを投入できますし、Elasticsearch Serviceもありますので、AWS上だけでも多くのことが可能です。しかし、アクティブユーザー数などビジネスレイヤーの監視を行うのであれば、Google Analyticsを使う場合が多いと思います。APMはフレームワークや言語との相性もあるので、New Relicを使った方がよい場合もあるでしょう。自分たちが確認したい値をAWSのサービスで表現できるか確認し、難しそうであれば無理せず、別のツールを使うというのが基本方針になります。AWSで完結することにはこだわらず、監視として適切な方式を選択できているかを重視します。

3.2 CloudWatchの進化

一方で、CloudWatchもここ数年で大きく進化しており、AWSのサービスのみでこと足るケースも増えてきました。最近の主な追加機能を列挙します。

- ・ダッシュボード機能
- ・メトリクスの保持期間の延長（15ヶ月）
- ・CloudWatch Agent
 - OS上の（EC2内の）メトリクス、ログ収集エージェント
- ・Metric Math
 - 複数のメトリクスに数式を適用して、新たなメトリクスを作成できる
- ・Container Insights
 - ECSで起動したコンテナの詳細監視

数年前であれば、ZabbixをメインにZabbixでは取れないメトリクス（RDSなど）をCloudWatchから取り込むといった、どちらかというと補助的な使い方が多かったですが、現在はCloudWatch

をメインに据えた監視設計が、十分可能になりました。特に、メトリクスの保持期間が延長したことが、大きく寄与しています。

すでに監視基盤がある場合は、他のツールをメインに使っていることが多いでしょう。私も仕事では他のツールをメインに、CloudWatchは補助的に使っています。ただ、今からゼロベースでAWS上の監視を設計する場合は、コストと可用性の部分でメリットが大きいので、CloudWatchをメインに設計していくと良いと思います。

3.3 AWSにおけるメトリクスの収集

AWSにおいては、ほとんどの項目については、AWSリソースを作成した時点でメトリクスの収集が開始されます。ですが、ユーザーが明示的に収集を行う必要がある部分もあります。この区分けはAWSの責任共有モデルに近いです。責任共有モデルでユーザーが責任を持つ部分に関しては、何らかのアクションを行う必要があります。AWSが責任を持つ部分は、メトリクスが提供されるといイメージです。

AWSの責任共有モデル
<https://aws.amazon.com/jp/compliance/shared-responsibility-model/>

3.4 CloudWatchの機能

AWSの監視で中心になるのはCloudWatchです。まずは、CloudWatchにどういった機能があるか、ざっと一覧化します。

ダッシュボードを使った運用を紹介

第5章 ダッシュボード

設計・実装した監視を運用していく際に、中心になるのはダッシュボードです。アラートが発火した際、最初に見に行くのはダッシュボードになり、ダッシュボードから異常のある箇所あたりを見つけ、ドリルダウンしていきます。また、アラートを設定するのは本当にクリティカルなものだけですが、アラートを設定しないメトリクスは何か障害が起きたときだけ見るかということ、そんなことはありません。アラートを設定しただけで、何か異常がないかは定期的に見る必要があります。サービスの稼働状況を徹底的に十分な情報量にまとめた、美しいダッシュボードを設計すると、運用の質が異なります。

5.1 ダッシュボードの設計

ダッシュボードの設計は、以下の作業になります。

- ・表示するメトリクスを選択する
- ・メトリクスの表現方法を定める
- ・表示順を決める

表示するメトリクスを選択する

全てのメトリクスをひとつのダッシュボードに表示しようとしてはいけません。特に今は大量にメトリクスの種類がありますので、全部表示すると情報過多で煩雑になり、誰も見ないダッシュボードができあがります。ポイントとなるメトリクスを選択し、1枚のダッシュボードにまとめます。実際に選択するメトリクスはサービスにより異なりますが、ピックアップする基準としては、以下と考えています。

- ・メトリクスの意味を理解できる
- ・過去の障害で原因となったことがある
- ・日常的に変化がある

メトリクスの意味を理解できる

ダッシュボードは人が使うツールですので、人が扱いやすいか、換えるかを重要視するべきです。理解できないメトリクスを表示していても、それは意味のない情報になります。収集できるメトリクスは大量にありますので、全てを理解している人は正直多くはないと思います。私もいまだによくわからないメトリクスがあります。給料をもらっているプロである以上勉強しろという話ですが、人間の能力には限界があります。理解できていないメトリクスは無理して表示せず、諦めましょう。ダッシュボードのスペースも無限ではありませんし、本当に必要なメトリクスであればいくつか学ぶ機会があります。そのときに意味を理解し、ダッシュボードに追加しましょう。

過去の障害で原因となったことがある

これは主に運用が始まってからの話になります。障害の原因になったメトリクスがある場合、まずはダッシュボードに表示して経過をみます。病気がケガが治ったあとにも、経過観察で医者に行くと思います。しかし、こういったメトリクスはその後変化がなければ、外してしまってもよいです。情報には鮮度があります。障害の後はホットなのでみんな興味を持ちますが、その後変化がないと興味を失うので、不要な情報になります。そうなれば、定期的に確認しても意味がないので外しましょう。

日常的に変化がある

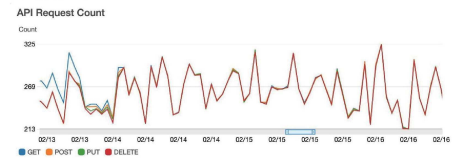
定期的なダッシュボードの確認で注視するのは、ひとつひとつのメトリクスの直近の値よりも、傾向です。グラフの形を視覚的に見て、普段と異なるものがないか見ていきます。特に、こういった傾向の変化はアラートを設定するのが難しく、今のところ実際に人の目で見ることが多いです。しかし、ずっと同じ値が続き変化がなく、直線になるようなメトリクスは表示しない方がよいです。変化がないと興味を失ってしまい、意味のない情報になっていくからです。変化があることが異常なのであれば、アラートを設定してSlack等に通知すれば十分でしょう。

メトリクスの表現方法を定める

表現方法はダッシュボードツールの表現力に依存しますが、基本的なものは以下です。

折れ線グラフ

図 5.1: 折れ線グラフサンプル



ダッシュボードにおけるメトリクスの表現で、一番よく使用するのは折れ線グラフです。監視メトリクスは連続した時系列データですので、折れ線グラフが一番表現しやすいです。

<<目次>>

第1章 監視の設計

第2章 SLI/SLO

第3章 AWS における監視設計

第4章 監視対象の多様化

第5章 ダッシュボード

第6章 AWS 上のサーバーレスアプリケーションの監視

付録A CloudWatch Synthetics を使った Frontend の監視

<<著者紹介>>

佐藤 瞬

福島県会津若松市出身。Slrとしてキャリアをスタートし、AWS 専門のインフラエンジニアとして、様々な Web システムの構築・運用を担当。その後、メーカーに身を移し、ユーザー企業として AWS でシステム開発・運用を行っている。

<<販売ストア>>

電子書籍:

Amazon Kindle ストア、楽天 kobo イーブックストア、Apple Books、紀伊國屋書店 Kinoppy、Google Play Store、honto 電子書籍ストア、Sony Reader Store、BookLive!、BOOK☆WALKER

印刷書籍:

Amazon.co.jp、三省堂書店オンデマンド、honto ネットストア、楽天ブックス

- ※ 各ストアでの販売は準備が整いしだい開始されます。
- ※ 全国の一般書店からもご注文いただけます。

【インプレス R&D】 <https://nextpublishing.jp/>

株式会社インプレスR&D(本社:東京都千代田区、代表取締役社長:井芹昌信)は、デジタルファーストの次世代型電子出版プラットフォーム「NextPublishing」を運営する企業です。また自らも、NextPublishingを使った「インターネット白書」の出版など IT 関連メディア事業を展開しています。

※NextPublishing は、インプレス R&D が開発した電子出版プラットフォーム(またはメソッド)の名称です。電子書籍と印刷書籍の同時制作、プリント・オンデマンド(POD)による品切れ解消などの伝統的出版の課題を解決しています。これにより、伝統的出版では経済的に困難な多品種少部数の出版を可能にし、優秀な個人や組織が持つ多様な知識の流通を目指しています。

【インプレスグループ】 <https://www.impressholdings.com/>

株式会社インプレスホールディングス(本社:東京都千代田区、代表取締役:松本大輔、証券コード:東証1部9479)を持株会社とするメディアグループ。「IT」「音楽」「デザイン」「山岳・自然」「モバイルサービス」「学術・理工学」「旅・鉄道」を主要テーマに専門性の高いメディア&サービスおよびソリューション事業を展開しています。さらに、コンテンツビジネスのプラットフォーム開発・運営も手がけています。

【お問い合わせ先】

株式会社インプレス R&D NextPublishing センター

TEL 03-6837-4820

電子メール: np-info@impress.co.jp