

記事

[Shintaro Kaminaka](#) · 2020年7月3日 8m read

組み込みREST APIを使用し、InterSystems IRISの監視

IRIS 2019.4以降の製品には、Prometheus形式でIRISのメトリックを公開する/api/monitorサービス機能が実装されています。

IRISのメトリック監視・警告ソリューションの一部として使用したい人にとっては大きなニュースです。

このAPIは、IRISの次期バージョンでリリースされる予定の新しいIRIS

System Alerting and Monitoring (SAM) ソリューションのコンポーネントです。

ただし、IRISインスタンスを監視するためにSAMがこのAPIの計画実証実験を開始するのを待つ必要はありません。今後の記事では利用可能なメトリックの意味についてさらに掘り下げ、対話型ダッシュボードの例を示します。しかし、まずは背景の説明はいくつかの質問回答から始めましょう。

IRIS(およびCaché)は常に自分自身の実行プラットフォームに関する数十のメトリックを収集しています。

[これらのメトリックを収集し、Caché/IRISを監視する方法は常に複数存在します](#)。また、IRISとCachéの組み込みソリューションを使用しているインストール環境はほとんどないことが分かっています。例えば、History Monitorはパフォーマンスシステムの使用状況に関するメトリックの履歴データベースとして長い間利用されてきました。

しかし、これらのメトリック計測システムをリアルタイムで表示する明確な方法はありませんでした。

IRISプラットフォームソリューションは(その他のあらゆるソリューションと同様に)少数のオンプレミスインスタンスで実行されている単一のモナリシックアプリケーションから「あらゆる場所」に配置される分散ソリューションに移行しています。既存のIRIS監視オプションは、多くの使用事例でこれらの新しいパラダイムに適合していません。InterSystemsは既存のものを作り直すのではなく、好評で実績のある現在の監視・警告用オープンソースソリューションに目を向けました。

Prometheusとは?

Prometheusは実績のある機に基いた、広く普及している有名なオープンソースの監視システムです。同システムにはさまざまなプラグインがあります。また、クラウド環境内で適切に機能するように設計されていますが、オンプレミスでも同様に役立ちます。プラグインには、オペレーティングシステム、ApacheなどのWebサーバ、および他の多くのアプリケーション向けのものがあります。PrometheusはGrafanaなどのフロントエンドクライアントでよく使用されており、高度なカスタマイズ機はUI/UXエクスペリエンスを提供します。

Grafanaとは?

Grafanaはオープンソースです。この連載を進める中で、一般的なシナリオの監視ダッシュボードのサンプルテンプレートをご紹介します。サンプルをベースに監視対象項目のダッシュボードを設計できます。本当の力は、状況に応じ、IRISメトリックを使用中のソリューションスタック全体メトリックを組み合わせる際に発揮されます。例えば、プラットフォームのコンポーネント、オペレーティングシステム、IRISのメトリックがありますが、特にアプリケーションからインストールメンテションを追加する場合です。

これらは実績のあるソリューションです。

PrometheusおよびGrafanaによるIRISとCachéの監視は目新しいことではありません。私はこれらのアプリケ

アプリケーションを数年間使用し、自分の開発環境やテスト環境を監視しています。開発者コミュニティで「Prometheus」を検索していただければ、CachéのメトリックをPrometheusで使用するために公開する方法を説明した他の投稿 ([Mikhail Khomenkoによるいくつかの優れた投稿など](#))が見つかります。

現在の違いは、/api/monitor APIが含まれ、デフォルトで有効になっていることです。メトリックを公開するために独自のクラスをコーディングする必要はありません。

Prometheus 入門

ここでは、Prometheusのいくつかの用語について簡単に説明します。私はIRISまたは他のソースが提供するメトリックの可視化方法や利用方法について、概要を理解し、礎を固め、実践する機会を作っていただきたいと考えています。

Prometheusは、アプリケーションがHTTPエンドポイント(IRIS /api/monitor などの種別API)として公開する時系列データをスライピングまたは取得することで機能します。エクスポートクライアントライブラリは多くの言語、フレームワーク、オープンソースアプリケーションに対応しています。例えば、ApacheなどのWebサーバ、オペレーティングシステム、Docker、Kubernetes、データベース、現在のIRISなどが挙げられます。

エクスポートはアプリケーションやサービスを計測し、スライピング用にエンドポイントに関連メトリックを公開するために使用されます。Webサーバ、データベースなどの標準コンポーネントは基本的なエクスポートでサポートされています。多くのエクスポートは、Prometheusコミュニティからオープンソースで公開されています。

Prometheus用語集

知っておくべき用語をいくつかご紹介します。

- ターゲットは、ホストやアプリケーション、ApacheやIRISなどのサービス、独自のアプリケーションなど、監視対象となるサービスの場所です。
- PrometheusはHTTPを利用してターゲットをスライピングし、メトリックを時系列データとして収集します。
- 時系列データは、IRISなどのアプリケーションやエクスポートを介して公開されます。
- エクスポートは、Linuxカーネルのメトリックなどの制御できないものに使用できます。
- 結果の時系列データは、Prometheusサーバのデータベース内にローカルに保存されます**。
- 時系列データベースは、最適化されたクエリ言語(PromQL)を使用して照会できます。例えば、アラートを作成したり、Grafanaなどのクライアントアプリケーションを使用してダッシュボードにメトリックを表示したりできます。

**ネタバレ注意: セキュリティ、スケーリング、高可用性および他の運用効率の理由から、新しいISA MソリューションではPrometheusの時系列データに使用されるデータベースはIRISになっています! ただし、IRISでのPrometheusデータベースへのアクセスは透過的であるため、Grafanaなどのアプリケーションがその事を認識したり、問題にすることはありません。

Prometheusのデータモデル

APIが返すメトリックはPrometheus形式です。Prometheusは1行ごとに1つのメトリックを持つ単純なテキストベースのメトリック形式を使用します。次のような形式になっています。

```
<identifier> [ (time n, value n), ....]
```

メトリックには、(キ、値)のペアとしてラベルを付けることができます。
ラベルは、メトリックをディメンションとしてフィルタリングする強力な方法です。例として、IRIS /api/monitorに対して返される単一のメトリックを調べます。
この場合、ジャーナルの空き容量は次のようになります。

```
iris_jrn_free_space{id="WIJ",dir="/fast/wij/"} 401562.83
```

識別子はそのメトリック名であることを表し、その出処を伝えるものです。

```
iris_jrn_free_space
```

複数のラベルを使用してメトリックを装飾し、後からフィルタリングがエリに使用できます。
この例では、WIJと、WIJが格納されているディレクトリを確認できます。

```
id="WIJ",dir="/fast/wij/"
```

また、値は401562.83 (MB) です。

どんなIRISメトリックを利用できますか？

[APIのドキュメント](#) のメトリックのリストを掲載しています。
ただし、リストの内容は変更される可能性がありますのでご注意ください。または、単に /api/monitor/metrics エンドポイントを照会し、リストを確認することができます。
コミュニティの次の投稿では、[Postman](#)の使い方についてご説明します。

何を監視対象にすべきですか？

システムとアプリケーションの監視方法を検討する場合は次の点に留意してください。

- 可能な場合は、ユーザに影響を与える重要なメトリックを計測してください。
 - ユーザはあなたが所有するマシンのCPU能力不足を気にしません。
 - ユーザはサービスの遅延やエラーを気にします。
 - プライマリダッシュボードでは、ユーザに直接影響する重要なメトリックに注目してください。
- ダッシュボードをグラフで埋め尽くさないでください。
 - 人間は一度に多くのデータを処理することはできません。
 - 例えば、サービスごとにダッシュボードを用意してください。
- マシンではなくサービスについて考えてください。
 - 問題を1つのサービスに特定したら、それを掘り下げ、1台のマシンに問題があるかどうかを確認できるようにします。

参考情報

[Prometheus](#) と [Grafana](#) のドキュメントとダウンロード

SAM (PrometheusとGrafanaを含む) のリリース概要をInterSystems Global Summit 2019で発表しました。
リンクは [InterSystemsの学習サービス](#) でご確認ください。

組み込みREST APIを使用し、InterSystems IRISの監視

Published on InterSystems Developer Community (<https://community.intersystems.com>)

直接リンク機能しない場合は、[InterSystems 学習サービスのWebサイト](#) にアクセスして「System Alerting and Monitoring Made Easy」を検索してください。

コミュニティで「Prometheus」や「Grafana」を検索してください。

次のパート「[例: デフォルトのREST APIを使用し、InterSystems IRISの監視メトリックの確認](#)

」を読み進めてください。

[#API](#) [#DevOps](#) [#オープンソース](#) [#システム管理](#) [#ダッシュボード](#) [#監視](#) [#InterSystems IRIS](#) [#InterSystems IRIS for Health](#)

ソースURL:

<https://jp.community.intersystems.com/post/%E7%B5%84%E3%81%BF%E8%BE%BC%E3%81%BFrest-api%E3%82%92%E4%BD%BF%E7%94%A8%E3%81%97%E3%81%9Fintersystems-iris%E3%81%AE%E7%9B%A3%E8%A6%96>