

文章

[Jingwei Wang](#) · 十二月 22, 2021 阅读大约需 5 分钟

使用内置REST API监控InterSystems IRIS

[IRIS 2019.4 预览版](#)中发布了/api/monitor服务，以Prometheus格式展示IRIS指标，但没有正式公布。对于任何想要使用IRIS指标作为其监控和警报解决方案的一部分的人来说，这是一个大新闻。该API是新的IRIS *系统警报和监控 (SAM)* 解决方案的一个组成部分，将在IRIS的一个即将到来的版本中发布。

然而，你不必等待SAM开始规划和试用这个API来监控你的IRIS实例。在未来的文章中，我将深入挖掘可用的指标和它们的含义，并提供交互式仪表盘的例子。但首先，让我从一些背景和一些问题和答案开始。

IRIS (和Caché) 总是在收集关于它自己和它所运行的平台的几十个指标。一直以来都有[多种收集这些指标的方法来监控Caché和IRIS](#)

。我发现，很少有安装使用IRIS和Caché内置的解决方案。例如，History Monitor仪表盘作为性能和系统使用指标的历史数据库已经存在很长时间了。然而，没有明显的方法来展现这些指标，并实时监测系统。

IRIS平台的解决方案正在从运行在几个企业内部的实例上的单体应用程序，转向分布式解决方案部署在“任何地方”。对于许多用例，现有的IRIS监控选项并不适合这些新的模式。InterSystems没有使用陈旧的方法，而是寻找当前流行的、经过验证的开源解决方案进行监控和警报。

普罗米修斯？

普罗米修斯是一个著名的、广泛部署的、技术成熟的开源监控系统。它有各种各样的插件，且能在云环境中很好地工作，对本地部署也同样有用。其插件包括操作系统、网络服务器（如Apache）和许多其他应用程序。普罗米修斯通常与可视化工具一起使用，例如，Grafana。

Grafana？

Grafana也是开源的。随着这一系列文章的进行，我将提供常见场景的监控仪表盘的样本模板。你可以把这些样本作为基础，为你所关心的事情设计仪表盘。当你把IRIS的指标与你整个解决方案堆栈的指标结合起来时，真正的力量就来了。从平台组件、操作系统、IRIS，特别是当你从你的应用程序中添加仪器时。

以前没有使用过吗？

用Prometheus和Grafana监控IRIS和Caché并不新鲜。几年来，我一直在使用这些应用程序来监控我的开发和测试系统。如果你在开发者社区搜索“Prometheus”，你会发现其他的帖子（[例如，Mikhail Khomenko的一些优秀帖子](#)），显示如何暴露Caché指标供Prometheus使用。

现在版本的区别是，/api/monitor API被包括在版本内，并且默认启用。不需要编码你自己的类来暴露度量。

普罗米修斯入门

这里是对Prometheus和一些术语的快速介绍。我希望你能看到高层次的东西，帮你考虑如何可视化或使用metrics打下一些基础。

普罗米修斯的工作方式是刮取或拉取从HTTP端点（API，如IRIS /api/monitor）的应用程序中暴露的时序数据。Exporters和客户端库存在于许多语言、框架和开源应用程序，例如，网络服务器Apache、操作系统、docker、Kubernetes、数据库以及现在的IRIS。

Exporters被用来检测应用程序和服务，并在一个端点上展示相关的指标，以便进行数据拉取。核心exporters支持网络服务器、数据库等标准组件。许多其他开源exporters可从Prometheus社区获得。

普罗米修斯术语

有几个关键术语是需要了解。

- Targets是服务所在的目标，如主机或应用程序或Apache或IRIS等服务或你自己的应用程序。
- 普罗米修斯通过HTTP对目标进行抓取，收集时间序列数据的指标。
- 时间序列数据是由应用程序暴露的，例如IRIS或通过exporters。
- Exporters可用于你无法控制的东西，如Linux内核指标。
- 由此产生的时间序列数据被储存在Prometheus服务器的本地数据库中。
- 时间序列数据库可以使用优化的**查询语言（PromQL）**进行查询。例如，创建警报或由客户应用程序（如Grafana）在仪表板上显示指标。

然而，对Prometheus数据库的访问--在IRIS上--是透明的，Grafana等应用程序不知道也不关心。

Spoiler

Alert

：出于安全、扩展、高可用性和其他一些运行效率的考虑，对于新的SAM解决方案，用于Prometheus时间序列数据的数据库是IRIS！这也是为什么我们要把Prometheus数据库的访问权放在IRIS上。然而，对Prometheus数据库的访问在IRIS上是透明的，Grafana等应用程序不知道也不关心。

普罗米修斯数据模型

由API返回的metrics是Prometheus格式的。普罗米修斯使用一种简单的基于文本的指标格式，每行有一个指标，格式是。

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

衡量标准使用标签（键，值）对。标签是一种强大的作为维度的方式来过度量。例如，检测IRIS /api/monitor返回的单一指标：日志的可用空间。

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

标识符告诉你该指标是什么。

```
iris_jrn_free_space
```

多个标签可以用来修饰指标，用来过滤和查询。在这个例子中，可以看到 WIJ 和存储 WIJ 的目录。

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

和值: 401562.83 (MB).

What IRIS metrics are available?

The [preview documentation](#) has a list of metrics. However, be aware there may be changes. You can also simply query the `/api/monitor/metrics` endpoint and see the list. I use [Postman](#) which I will demonstrate in the next community post.

可以用哪些IRIS指标？

[预览文档](#)

有一个metrics的列表。请注意可能会有变化。你也可以简单地查询/api/monitor/metrics端点并查看该列表。我将在下一篇社区文章中演示使用[Postman](#)调用这个端点。

应该监控什么？

当你考虑如何监控你的系统和应用程序时，请牢记这些要点。

- 对影响用户的关键指标进行检测。
 - 用户并不关心你的一台机器的CPU是否不足。
 - 用户关心的是服务是否缓慢或有错误。
 - 对于你的主要仪表盘来说，重点是直接影响用户的高级指标。
- 对于你的仪表盘来说，要避免一整面墙的图表。
 - 人类无法同时处理太多的数据。
 - 例如，每个服务拥有一个仪表板。
- 考虑服务，而不是机器。
 - 一旦你把问题隔离到一个服务上，那么你就可以深入研究，看看是否是一台机器的问题。

参考资料

文档下载：[Prometheus](#) 和 [Grafana](#)

我在InterSystems

2019年全球峰会上介绍了SAM（包括

Prometheus和Grafana）的预发布概述，你可以找到[InterSystems学习服务网站的链接](#)

。如果链接不起作用，请进入[InterSystems学习服务网站](#)并搜索。"系统警报和监控变得简单"

在社区中搜索 "Prometheus" 和 "Grafana"。

[查看原帖](#) 由 [@Murray Oldfield](#) 撰写

[#系统警报和监视 \(SAM\)](#) [#InterSystems IRIS for Health](#)

源

URL:<https://cn.community.intersystems.com/post/%E4%BD%BF%E7%94%A8%E5%86%85%E7%BD%AErest-api%E7%9B%91%E6%8E%A7intersystems-iris>