Journal of Technological Innovations

Est. 2020

APPLICATION PERFORMANCE MONITORING FOR END-TO-END TRANSACTION PERFORMANCE

Naveen Muppa
Email: naveen.muppa1@wipro.com

Abstract:

Consider a business transaction to be a user-generated action within your system. The best practice for determining the performance of your application isn't to measure CPU/Memory usage, but to track the flow of a transaction that your customer, the end user, has requested.

Keywords: AppDynamics is an Application Performance Monitoring (APM) tool that allows you to monitor your end-to-end transaction performance. Once you have deployed your application into the production environment, it is a recommended best practice to monitor the end-to-end performance of your application, identify and resolve production issues and integrate with Operations management. AppDynamics can help mitigate these issues on hand. AppDynamics ties into the tenets of application reliability, namely: Awareness, Escalation, Triage, Remediation and Action. Hence, we do recommend the AppDev teams to install, setup and configure AppDynamics for your environment.

1. INTRODUCTION

AppDynamics provides application context which the built-in tools may not provide. It also monitors real-time, analyzes slowness in your application, sends alerts proactively, and has drilldown ability from the application level to the database level. You can perform trend analysis of JVM JMX Metrics and MBeans in real-time and assess how resources inside your JVM are being consumed by application requests.

2. APPDYNAMICS: SETUP, CONFIGURATION ROLES & RESPONSIBILITIES

AppD Installation

The steps to setup Application Monitoring for AppD are:

- Go to the homepage and Type keyword: "AppD"
- Click on "Java Technology"
- Click on "Order and Download Software"
- Click on "AppDynamics"

- You can click on "Automated Architecture Delivery Install", if you utilize EBR (recommended) OR click on "Manual Install" and follow the directions from there onwards.

Standard Process for Java Applications

- These will be the steps to install AppD for most Java-based applications.
- AppDynamics is already installed and configured.
- Check the folder /opt/appd/ to verify that it is installed.
- Your application user must be a member of the AppD sudo group. Contact AppDynamics TFA for assistance.
- Non-CloudOps VMs or Physical Hardware running Linux.
- Check if AppDynamics is installed in the folder /opt/appd/
- Contact SA team and AppDynamics team to ensure that correct version is installed and ensure that your application user is a member of the appd sudo user group on the host machine.

AppDynamics Terminology

The AppDynamics setup at is conventionally different if you have worked with other APM tools such as Wily/Introscope. Kindly pay attention to the terminologies below as they will be referenced throughout this guide.

- Application: The largest granularity at which you would want to manage the performance of your environment is defined as an “Application” in the AppDynamics model. An “application” in AppDynamics model is essentially a conglomerate of “*individual applications*” as we know conventionally know.

- Tier: An AppDynamics tier represents an instrumented service (such as a web application) or multiple services that perform the.

3. MONITORING AND TRENDING

AppDynamics monitors every single business transaction (BT) that flows through the application. It automatically learns and develops a dynamic baseline (historical data collected over a time to compare with current performance) for end-to-end response time, as well as the response time of every component along the transaction flow. It scores each transaction by comparing the actual response time to the self-learned baseline. When it determines that a BT has deviated too far from normal behavior, it automatically collects full call stack details for troubleshooting.

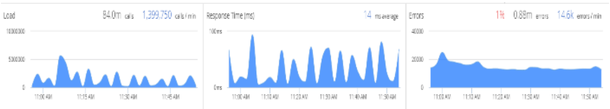


Figure 1. Monitoring and Trending

4. ALERTING AND THESHOLDS

Alerts enable us to define boundaries around acceptable performance and also setup notifications if performance falls below the acceptable limit using either percentage or standard deviation measurements. Operational alerts can be setup via email or SMS. The context of the notification is automatically generated by the event that triggered the action. We can classify the alerts based on custom transaction thresholds, we set up for the apps such as: Slow, Very Slow or Stalled.

The default thresholds are:

Slow = 3 x standard deviation

Very Slow = 4 x standard deviation

Stalled = 45000 ms

Threshold definitions can be viewed under Configuration > Slow Transaction Thresholds. If the default settings don’t meet your need, you can custom define them to tailor to your environment. We can also set up health rules to send alerts based on custom defined critical or warning condition.

Q. Who should be the recipient of the Warning Alerts? App-Dev Group, Testing Group.

Q. Who should be the recipient of the Critical Alerts? App-Dev Lead, Operations Group

5.INSTUMENTATION AND KPI’S

Instrumenting your application provides a measurement of your application's performance in real time. This measurement tells us how long the application took to execute and identifies bottlenecks. Dynamic versus Custom Instrumentation. Dynamic Instrumentation: AppDynamics will automatically detect all the business transactions and entry points that it deems as critical to your application. However, if AppDynamics does not accurately map to each of the business transactions /entry points or critical processing components, then it provides the option of Custom Instrumentation. Custom instrumentation allows the developers to map and customize the entry points based on custom match rules or dynamic values. The Developer needs to exclude/include the transactions they need to monitor based on class/method combination, custom match rules, based on functionality, transactions that get the most traffic etc. In other words, define the criteria most significant to application needs.

Example:

Class:

com..srs.edd.customerdiscountflags.access.svc.impl. EddCustomerDiscountFlagsServiceImpl

Methods:

getCustomerDiscountFlag - Endpoint

getCachedPricing - Get pricing from cache

getEddPricing - Gets pricing flags from eDD database

getLegacyPricing - Gets pricing flags from Cheers Pricing

Class:

com..srs.edd.getpricing.domesticrating.dao.Domestic RatingDao

Methods:

retrievePricingFlags

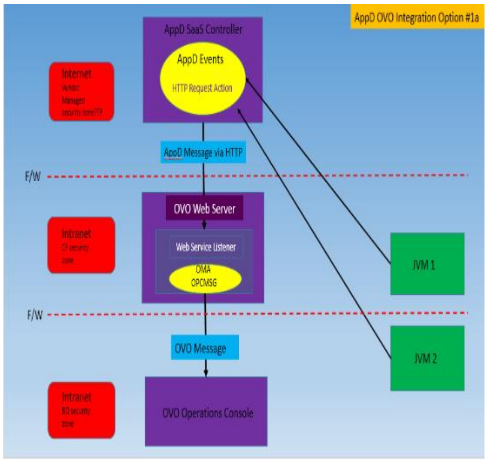
Note: The methods are automatically detected by AppD based on the built-in auto discovery feature. However, the critical methods need to be identified by the developers.

These methods appear as part of a Business Transaction/ Information Point /Service End Point and can be used for Monitoring and Instrumentation purposes within the Metrics Browser to analyze statistics or create a custom dashboard.

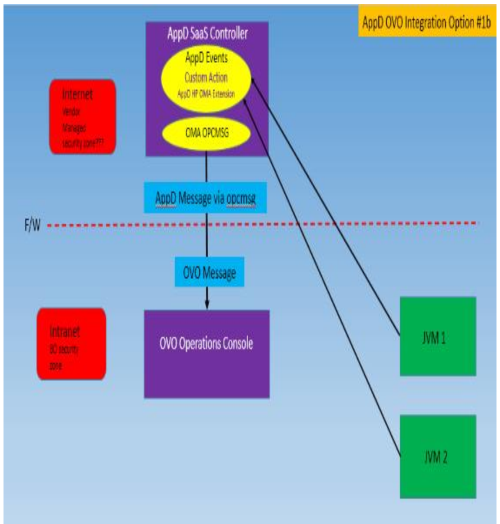
6.APPD-OVO ALERTS INTEGRATION

Here are the 3 proposed solutions for AppD-OVO Integration:

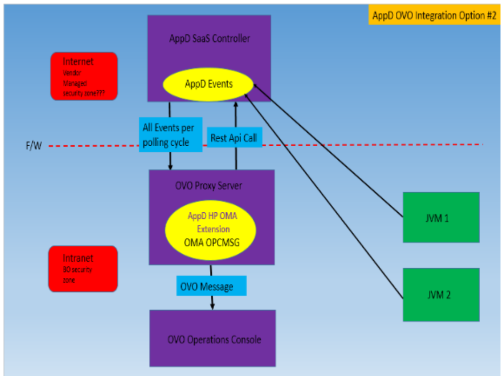
Solution1:



Solution2:



Solution3:



.

7.CONCLUSION

This section on monitoring garbage collection is for informational purposes. The developer need not make configuration changes to monitor this activity. App Dynamics provides out-of-the-box metrics for monitoring garbage collection statistics.

AppDynamics captures performance data about the duration of the pauses for garbage collection. These are usually instrumented using JMX. In other words, JMX provided built-in instrumentation for your applications. AppDynamics automatically discovers JMX attributes. JMX uses objects called MBeans (Managed Beans) to expose data and resources from your application.

Below are some of the graphs and stats AppD provides:

Below the Heap panel, the Garbage Collection - Time Spent panel shows how much time, in milliseconds, it takes to complete both minor and major collections.

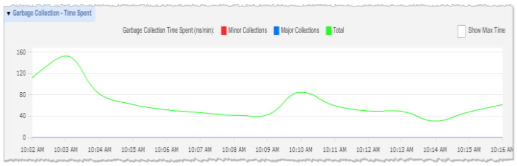


Figure 2. Garbage Collection Time spent.

The Garbage Collection - Minor Collections panel shows the number of minor collections per minute. The effectiveness of minor collections indicates better performance for your application.

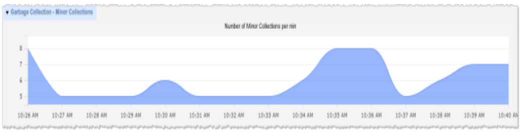


Figure 3. Garbage Collection - Minor Collections

The Garbage Collection - Major Collections panel shows the number of major collections per minute.

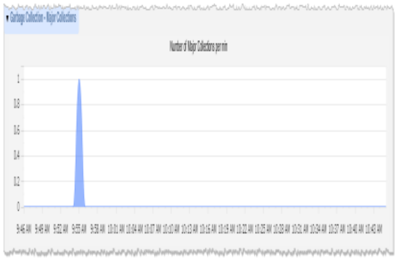


Figure 4. Garbage Collection - Major Collections

In the JMX Metrics subtab metric tree, you click an item and drag it to the line graph to plot current metric data.

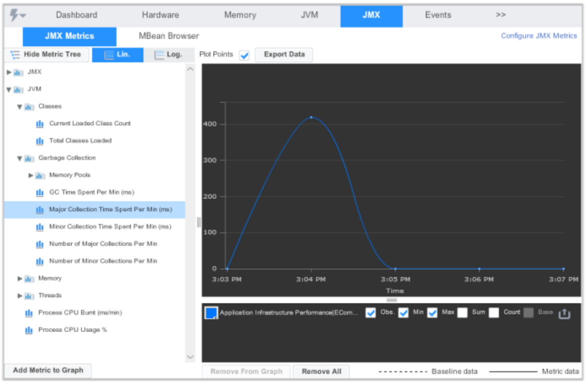


Figure 5. Garbage Collection Metrics

8.REFERENCES

[1] AppDynamics 101: Tips for Beginners, Online Tutorial, Demo http://www.appdynamics.com/learningbytes/

[2] AppDynamics Documentation https://docs.appdynamics.com/display/DASH/Learn+AppDynamics

[3] AppDynamics Best Practices for Developers and Operations Professionals https://docs.appdynamics.com/display/ACE/AppDynamics+Best+Practices