Concurrency Quota Policy Usage Scenarios for Policy Manager

SOA | software™

Copyright

Copyright © 2015 SOA Software, Inc. All rights reserved.

Trademarks

SOA Software, Policy Manager, Portfolio Manager, Repository Manager, Service Manager, Community Manager, SOA Intermediary for Microsoft and SOLA are trademarks of SOA Software, Inc. All other product and company names herein may be trademarks and/or registered trademarks of their registered owners.

SOA Software, Inc.

SOA Software, Inc.
12100 Wilshire Blvd, Suite 1800
Los Angeles, CA 90025
(866) SOA-9876
www.soa.com
info@soa.com

Disclaimer

The information provided in this document is provided "AS IS" WITHOUT ANY WARRANTIES OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF INTELLECTUAL PROPERTY. SOA Software may make changes to this document at any time without notice. All comparisons, functionalities and measures as related to similar products and services offered by other vendors are based on SOA Software's internal assessment and/or publicly available information of SOA Software and other vendor product features, unless otherwise specifically stated. Reliance by you on these assessments / comparative assessments is to be made solely on your own discretion and at your own risk. The content of this document may be out of date, and SOA Software makes no commitment to update this content. This document may refer to products, programs or services that are not available in your country. Consult your local SOA Software business contact for information regarding the products, programs and services that may be available to you. Applicable law may not allow the exclusion of implied warranties, so the above exclusion may not apply to you.
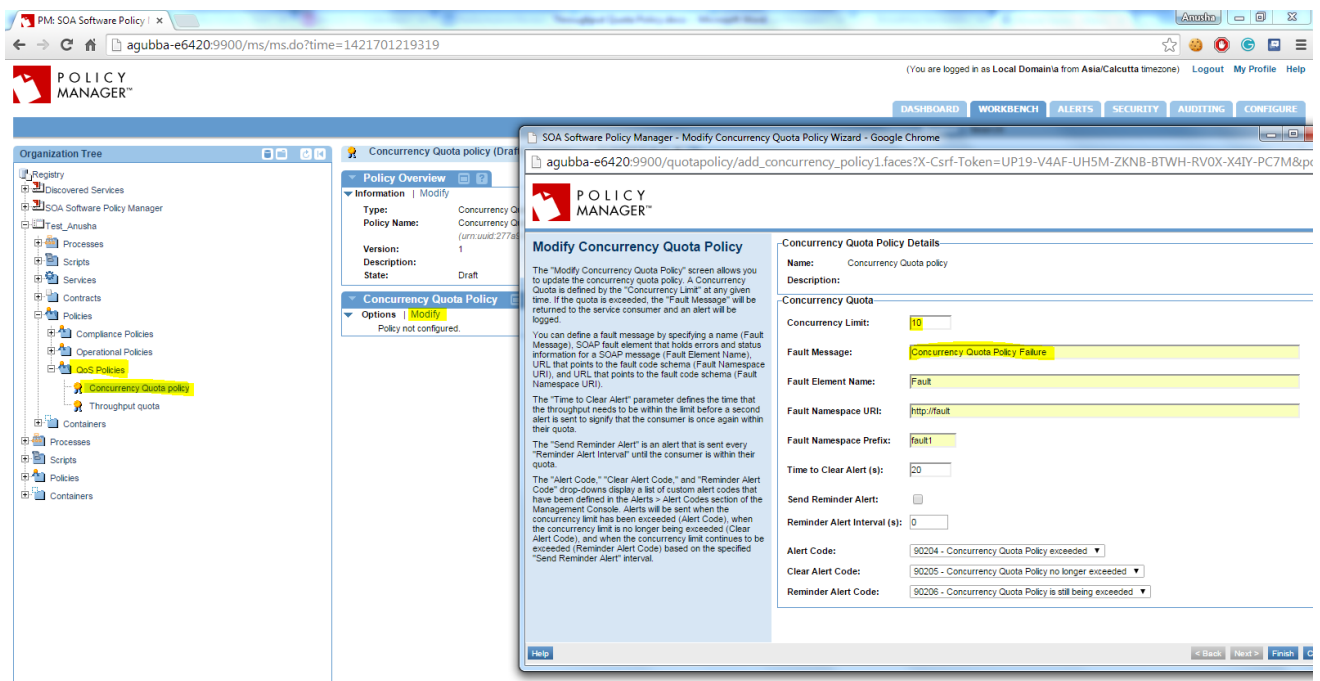
Contents

Concurrency Quota Policy Usage Scenarios (Policy Manager-specific)

This document provides a list of Policy Manager-specific usage scenarios for the *Concurrency Quota Policy*.

Scenario 1: Generate Alert When Concurrent Requests Received

Generate an alert and send an email to the administrator(s) if more than 10 concurrent requests are received by a service.

1   Create a physical service in *Policy Manager Management Console* using **Create Physical Service**.

2   Provide service details and **Finish** the wizard.

3   Using **Virtualize Service**, virtualize and host the physical service on Network Director (**ND1**), and assign a name (e.g., **Vs1**).

4   Navigate to *Organization > Policies > QOS Policies* and use **Add Policy** to create a *Concurrency Quota Policy*.

5   Configure the *Concurrency Quota Policy* as per the use case with 10 Concurrency Limit and a custom Fault message.

6   Activate the policy in the *Policy Workflow* portlet, and attach it to the **Vs1** service in the *Service Details > Policy Attachments > QoS* section.

    a) Send 20 concurrent requests from application/client to **Vs1** service.

    b) When the concurrency limit has exceeded, the Fault Message will be returned to the service consumer and an alert will be logged at service level.

7   The usage logs' details show the error being displayed as configured in the policy.

8   In the alert code, email has to be configured so as to send an email to the administrator(s) as below

Timeline
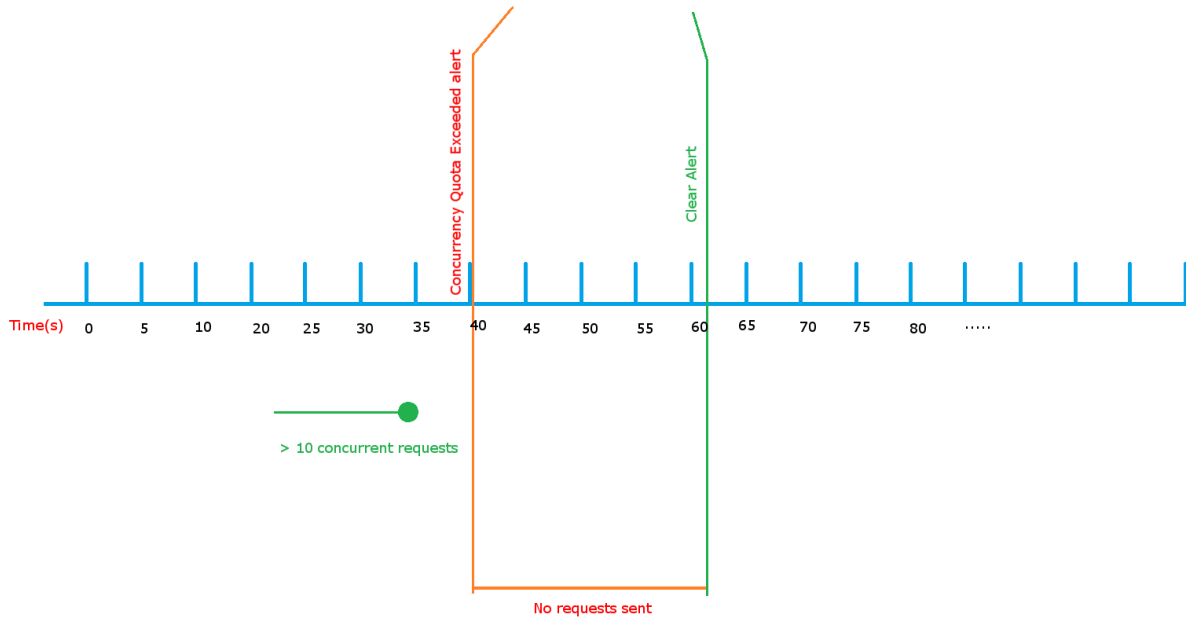
Scenario 2: Generate Reminder Alert When Concurrent Requests Received

Generate a remainder alert and run a Management Script if more than 10 concurrent requests per 10 seconds are received repeatedly by a service in the 100 seconds.

1   Create a physical service in *Policy Manager Management Console* using **Create Physical Service**.

2   Provide service details and **Finish** the wizard.

3   Using **Virtualize Service**, virtualize and host the physical service on Network Director (**ND1**), and assign a name (e.g., **Vs1**).

4   Navigate to *Organization > Policies > QOS Policies* and use **Add Policy** to create a *Concurrency Quota Policy*.

5   Configure the *Concurrency Quota Policy* as per the use case with 10 Concurrency Limit and a custom Fault message. Set Remainder Alert interval to 10 seconds and Clear alert time interval to 100 seconds.

6   Activate the policy in the *Policy Workflow* portlet, and attach it to the **Vs1** service in the *Service Details > Policy Attachments > QoS* section.

7   Send 20 concurrent requests from application/client to Vs1 service

    a)   When the concurrency limit has exceeded, the Fault Message will be returned to the service consumer and an alert will be logged at service level.

    b)   The usage logs' details show the error being displayed as configured in the policy.

8   Keep sending 20 concurrent requests from the application/client to Vs1 service.

    a)   During the first 10 seconds no alert is shown.

    b)   If the concurrency hits are seen after 10 seconds, a remainder alert is thrown which is valid for next 10 seconds.

    c)   After the first interval of 10 seconds, if the concurrency hits are still observed, again a remainder alert is seen.

d) Within the clear interval, for every 10 seconds, this alert is thrown if validation crosses

e) In the Alert code, a script can be configured to run (e.g., to store any sort of information related to the alert).

Scenario 3: Generate Alert When Using Contracts

Generate an alert if an application hits a contract when sending more than 15 concurrent requests to various services using the contract

1   Create two physical services in *Policy Manager Management Console* using **Create Physical Service**.

2   Provide service details and **Finish** the wizard.

3   Using **Virtualize Service**, virtualize each physical service and host on Network Director (**ND1**), and assign a name (e.g., **Vs1**, **Vs2**).

4   Navigate to *Organization > Policies > QOS Policies* and use **Add Policy** to create a *Concurrency Quota Policy*.

5   Configure the Concurrency Quota policy as per the use case with 15 Concurrency Limit and a custom Fault message.

6   Activate the policy in the *Policy Workflow* portlet, and attach the *Concurrency Quota Policy* to a contract which serves **Vs1** and **Vs2** services.

7   Send 25 concurrent requests from application/client to Vs1 and Vs2 services all together

    a)   When the concurrency limit has exceeded, the Fault Message will be returned to the service consumer and an alert will be logged at contract level.

    b)   The usage logs' details show the error being displayed as configured in the policy.

Scenario 4: Generate Clear Alert When Concurrency Exceeds Specified Interval

Generate a clear alert and send an email to the administrator(s) when concurrency is no more exceeded in the 100 seconds interval

1   Create a physical service in *Policy Manager Management Console* using **Create Physical Service**.

2   Provide service details and **Finish** the wizard.

3   Using **Virtualize Service**, virtualize and host the physical service on Network Director (**ND1**), and assign a name (e.g., **Vs1**).

4   Navigate to *Organization > Policies > QOS Policies* and use **Add Policy** to create a *Concurrency Quota Policy*.

---

5   Configure the *Concurrency Quota Policy* as per the use case with 10 Concurrency Limit and a custom Fault message.

6   Activate the policy in the *Policy Workflow* portlet, and attach it to the **Vs1** service in the *Service Details > Policy Attachments > QoS* section.

7   Send 20 concurrent requests from application/client to **Vs1** service.

   a)   When the concurrency limit has exceeded, the Fault Message will be returned to the service consumer and an alert will be logged at service level.

   b)   The usage logs' details show the error being displayed as configured in the policy.

   c)   In the next 100 seconds, if the policy is not violated, a clear alert is sent.

   d)   In the clear alert code, an email must be configured that will be sent to the administrator(s) as illustrated below:

Timeline (Scenarios 2 and 4)