Performance Tuning And Deployment Parameters

This chapter explains the various parameters that should be modified to obtain the best performance from FioranoMQ. This chapter also provides information on deploying FioranoMQ and additional software components provided with FioranoMQ in addressing various scaling and clustering issues.

Performance Tuning Parameters

The default configuration of FioranoMQ provides optimum performance under most traffic loads and for most common message sizes (ranging from 0-10 KB).

FioranoMQ administrators can tune the configurations, below, to optimize the performance of the FioranoMQ Server.

PTP Configuration Parameters

FioranoMQ provides the configuration parameters, below, to tune the point-to-point messaging model:

In-Memory Persistent Message Buffer

FioranoMQ server buffers the persistent messages in its in-memory cache besides storing these messages in the persistent store. This buffering is done to ensure fast delivery of messages to the listening receivers. The variable named **PersistentInMemoryBufferSize** depicts the size of the in-memory buffer whose default value is 512KB (512*1024). This value can be increased when the message send rate is high. This is because when the send rate is high, the buffer gets filled up faster than when the send rate of messages is lower.

The steps below enable the configuration of the In-Memory Persistent Message Buffer parameter using the Fiorano Admin Studio in online mode:

1. Invoke the Fiorano Admin Studio and login to the FioranoMQ Server through **JMX** login.

2. Select **Fiorano** > **mq** > **ptp** > **PtPManager** > **QueueingSubSystem** > **config** from the Server Explorer pane.

3. In the **Properties** pane, type in the new value against the property named **PersistentInMemoryBufferSize** and press Enter.

In-Memory Non-Persistent Message Buffer

The FioranoMQ Server stores the published non-persistent messages in an in-memory table. The size of this buffer is monitored by the parameter **NPInMemoryBufferSize** whose default value is 1MB. This buffer can gets full if the message send rate is quite high as compared to the message receive rate. When this buffer fills up, the FioranoMQ senders cannot push new messages into the buffer and messages remain blocked till a receiver is activated on the concerned queue to empty the buffer.

It is recommended that the size of the buffer is set to a value where chances of it becoming full are minimized, particularly in cases where the senders are sending messages at a much faster rate than the rate at which they are being received.

The steps below enable the configuration of the In-Memory Non-Persistent Message Buffer parameter through Fiorano Admin Studio in online mode:

1. Through the Fiorano Admin Studio login to the FioranoMQ Server using**JMX** login.

2. Select **Fiorano** > **mq** > **ptp** > **PtPManager** > **QueueingSubSystem** > **config** from the **Server Explorer** pane.

3. In the **Properties** pane, type in the new value of the property named **InMemoryBufferSize** and press Enter.

Prefetch Count

Prefetch count is the number of messages requested by a client from the Server in one 'receive call'. Prefetch count can be set through the **PrefetchCount** parameter using Admin Studio in the offline mode. The default value of this variable is 3. PTP prefetching enables the FioranoMQ Server to deliver a maximum number of prefetch count messages to the receiver on each receive call.

Perform the steps below to configure the Prefetch Count parameter:

1. Open the Fiorano Admin Studio.

2. Select **Tools > Configure Profile** from the menu bar. Select the **FioranoMQ** folder, and click the **Open** button. FioranoMQ is now in offline mode.

3. Navigate to **FioranoMQ -> Fiorano -> mq -> ptp -> QueueingSubSystem** in the **Profile Manager** pane. The properties of the PTP Manager are displayed in the **Properties** pane.

4. Select the **ellipsis** against the parameter **PrefetchCount**. Type in the new value in the **PrefetchCount** dialog box and click **OK**.

5. Right-click on the FioranoMQ domain in the **Server Explorer** pane and select the **Save** option from the shortcut menu.

Prefetch Threshold

Prefetch threshold denotes the minimum number of messages in the local buffer of the client that trigger a request for delivery of more messages to the client. Prefetch threshold can be set through a parameter named **PrefetchThreshold** using the Fiorano Admin Studio in the offline mode.

The default value of the Prefetch Threshold variable is 1. The number of messages that are requested from the server when the threshold limit is reached is equal to the prefetch count.

Perform the steps below to configure the Prefetch Threshold parameter:

1. Open the Fiorano Admin Studio.

2. Select **Tools > Configure Profile** from the menu bar. Select the **FioranoMQ** folder, and click the **Open** button. FioranoMQ is now in offline mode.

3. Navigate to **FioranoMQ -> Fiorano -> mq -> ptp -> QueuingSubSystem** in the **Profile Manager** pane. The properties of the PTP Manager are displayed in the **Properties** pane.

4. Select the **ellipsis** against the parameter **PrefetchThreshold**. Type in the new value in the **PrefetchThreshold** dialog box and click **OK**.

5. Right-click on the FioranoMQ domain in the **Profile Manager** pane and select the **Save** option from the shortcut menu.

PTP Prefetch size

The PTP Prefetch size parameter is used to specify the total size of messages (in bytes) that will be sent by the FioranoMQ Server on each receive call made by the client. Prefetch size can be set through a parameter named **MaxPrefetchSize** using the Fiorano Admin Studio in the offline mode. The default value of this variable is 512 KB. If both Prefetch Size and Prefetch Count are specified,the number of messages sent by FioranoMQ server is controlled by the minimum value among these two parameters.

Perform steps below to configure the Prefetch Size parameter:

1. Open Fiorano Admin Studio.

2. Select **Tools > Configure Profile** from the menu bar. Select the **FioranoMQ** folder and click the **Open** button. FioranoMQ is now in offline mode.

3. Navigate to **FioranoMQ -> Fiorano -> mq -> ptp -> QueuingSubSystem** in the **Profile Manager** pane. The properties of the PTP Manager are displayed in the **Properties** pane.

4. Select the **ellipsis** against the parameter **MaxPrefetchSize**. Type in the new value in the **MaxPrefetchSize** dialog box and click **OK**.

5. Right-click on the FioranoMQ domain in the **Profile Manager** pane and select the **Save** option from the shortcut menu.

Queue Sender Blocking Interval

This parameter represents the period, in milliseconds, for which the sender will be blocked (while it is unable to push more messages in the In-Memory Buffer maintained by the FioranoMQ Server). The parameter **SenderBlockingInterval** can be set through the Fiorano Admin Studio in online mode. The default value of this variable is 10 milliseconds. The Sender Blocking Interval parameter is used only where persistent messages are present. Setting this parameter to -1 allows a sender to continuously publish messages without blocking them. The Sender Blocking Interval is ignored if no receivers are registered on a queue.

Perform the steps below to configure the Sender Blocking Interval parameter:

1. Open Fiorano Admin Studio and login to the FioranoMQ server.

2. Select **Destinations > Queues** sub-node from the **Server Explorer** pane.

3. Select the Queue name whose properties are to be displayed.

4. In the **Properties** pane, type in the new value of **SenderBlockingInterval** and press Enter.

PubSub Configuration Parameters

FioranoMQ offers a very high message throughput and low latency for non-persistent messages. The boost in delivery rates have been achieved through a series of algorithmic changes for multi-threaded scenarios, data-copy optimizations, transport layer enhancements, changes in flow-control semantics and by removing bottlenecks from the code.

FioranoMQ provides the following configuration options for tuning the publish-subscribe model:

Setting the Message Receipt Acknowledgement

In the **DupsOkBatchSize** mode, acknowledgement of receipt of message is sent after a configurable number of messages. By default this number is 20.

The steps below enable configuration of the Message Receipt Acknowledgement parameter using the Fiorano Admin Studio in offline mode:

1. Open Fiorano Admin Studio.

2. Select **Tools > Configure Profile** from the menu bar. Select the FioranoMQ folder, and click the **Open** button. FioranoMQ is now in offline mode.

3. Navigate to **FioranoMQ -> Fiorano -> mq -> pubsub -> TopicSubSystem** in the **Profile Manager** pane. The properties of the CommonConfig are displayed in the **Properties** pane.

4. Select the **ellipsis** against the parameter **DupsOkBatchSize**. Type in the new value in the **DupsOkBatchSize** dialog box and click **OK**.

5. Right-click on the FioranoMQ domain in the **Profile Manager** pane and select the **Save** option from the shortcut menu.

Setting In-Memory Buffers for Subscribers

Another mechanism to manage flow-control is to increase the size of the In-Memory Buffers of topics and of subscribers with the Publisher slowdown algorithm option enabled. This will delay the initialization of publisher slowdown algorithm and increase the overall throughput. When the subscriber receives messages slowly, increasing the buffer size may lead to the Publisher slowdown algorithm not initializing. This depends on the rate at which the messages are being published on the server. This is configurable using **BasicAllowedSize** parameter through the Fiorano Admin Studio in offline mode. The default value of this variable is 128KB. The default value can be increased to a higher value depending on the scenario.

Following steps enable you to configure this parameter:

1. OpenFiorano Admin Studio.

2. Select **Tools > Configure Profile** from the menu bar. Select the FioranoMQ folder, and click the **Open** button. FioranoMQ is now in offline mode.

3. Navigate to **FioranoMQ -> Fiorano -> mq -> pubsub -> TopicSubSystem** in the **Profile Manager** pane. The properties of the PubSub Manager are displayed in the Properties pane.

4. Select the **ellipsis** against the parameter **BasicAllowedSize**. Type in the new value in the **BasicAllowedSize** dialog box and click **OK**.

5. Right-click on FioranoMQ domain in the **Profile Manager** pane and select the **Save** option from the shortcut menu.

Setting Parameters for New Pubsub Algorithm to Handle Slow Subscribes

In previous releases of FioranoMQ, there were no options available to disable the slowing down of the publisher when subscribers slow. In the default configuration, the publisher blocks messages by a predefined timeout and flow-control is activated.

From FioranoMQ SP2 onwards, the option of persisting messages to disk when the subscriber slows down, results in the fast subscriber not being affected even when publisher slowdown is disabled.

Below are the parameter that determines the different behaviors of the pubsub algorithm. Parameters can be configured from Fiorano Admin Studio (**FioranoMQ->fiorano->mq->pubsub->TopicSubSystem**).

Note: Pubsub parameters can be configured at either Topic level or at Subsystem level. Values taken from the topic level will be given higher priority over those taken from the Subsystem level. However, if the values at the topic level are set to the default value, then the value is fetched for the corresponding parameter from the Subsystem level itself. To set values at Topic level navigate to **FioranoMQ > Fiorano > mq > pubsub > Topics > TopicName** and change the corresponding parameters.

- **PublishBackoffThreshold:** This value indicates the size factor after which the Publisher needs to slow down. If its value is 0.6 and MaxPersistentSize is 1000MB, once the PSQ Size crosses 0.6*1000MB = 600MB then the Publisher will slow down.

  Navigate to::

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem->PublishBackoffThreshold.

By default this value is 0.6 which means 60% of PSQ.

- **StoreMessageToDisk**: When the session buffer in the client runtime overflows, the session state in the Server becomes passive. Thepublisher starts pushing the message into temporary persistent queues in the disk (for that particular session only). The slow subscriber will not receive messages until the client's session buffer can accommodate more messages. A normal subscriber keeps receiving the message since the publisher does not slow down.

  Navigate to: :

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem->StoreMessageToDisk

  The default value of this parameter is set to **yes**. The result of this parameter is not effective for non-persistent message that can be discarded.

  If the default parameter is set to **no**, this parameter ensures that messages within the FioranoMQ Server never **overflow** onto a disk queue. This means that the messages are either delivered to the subscriber (assuming the subscriber is fast enough to pick them up), or they are dropped (once the internal buffer fills up).

- **PublishWait:** indicates the default time interval by which the publisher slows down when **EnablePublisherSlowdown** is set to **yes**. As the size of the Persistent Session Queue (PSQ) changes, the publisher blocking time also changes according to the exponential back-off algorithm.

  Navigate to: :

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem->PublishWait

  The default value of this parameter is 50 milliseconds (values less than or equal to 0 are not accepted).

- **EnablePublisherSlowdown***:* If it's set to **no**, then the publisher does not slow down when sending messages to the persistent queue (PSQ). EnablePublisherSlowdown is used when publisher flow-control is needed due to many slow subscribers. This option should be carefully selected considering the different options as mentioned above. If this option is selected the publisher will slow down if PSQ size crosses the threshold defined by PublishBackoffThreshold. as described in the description of backoff algo. The slow down time starts from the value specified by another parameter called **publishWait**. It increases according to the slow down algorithm which depends on the size of the persistent message queue for every slow subscriber. This slow down time dynamically changes as and when persistent storage size changes. This option dynamically balances publisher slow down with the size of persistent message in the disk.

  Navigate to:

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem->EnablePublisherSlowdown

  The default value of this parameter is **yes**. If the User does not want to have publisher flow-control, this parameter should be set to **no**.

- **DropOldestMessag***e:* If set to **yes**, the publisher first drops the oldest message from the persistent store and pushes the latest on the queue thereby ensuring the maximum size of persistent store is kept at all times.

  Navigated to:

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem->DropOldestMessage

The default value of this parameter is set to **no.** If the user wants to drop the latest message, when the size of total messages in the disk cross over a certain threshold, then this parameter should be set to its default value.

- **MaxPersistentStoreSize:** Indicates the maximum size of total messages stored in the persistent queue in the disk. Once the current size crosses over this threshold, the publisher starts dropping them if DropOldestMessage flag is set to **no**. If this flag is set to **yes**, the publisher first drops the oldest message from the persistent store and pushes the latest one. If it's value is set to **-1**, messages will be persistent till the server run out of disk space.

  Navigated to:

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem->MaxPersistentStoreSize

  The default value of this parameter is 1073741824 bytes (1 GB).

- **PublishWaitTimeChunks:** If PSQ size reaches 95% of the maximum value, the publisher publishing persistent messages will be blocked till PSQ size reduces to 85%. The publisher should wait and periodically check the PSQ sizeto confirm it has reduced. This time interval at which publisher checks the PSQ can be controlled by this parameter.

  Before checking the PSQ size, publisher will wait for MessagePublishTimeout/ PublishWaitTimeChunks in milliseconds for each iteration.

  Navigated to:

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem-> PublishWaitTimeChunks

  The default value of this parameter is 8 (values less than or equal to 0 are not accepted).

- **IgnorePSQSizeForPersistentMessages:** This flag determines the behavior when PSQ size exceeds MaxPersistentStoreSize. If this flag is set to **true**, persistent messages will still go onto the PSQ.

  Navigated to:

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem-> IgnorePSQSizeForPersistentMessages

  The default value of this parameter is **Yes**.

- **MaxPublisherBlockTime:** When EnablePublisherSlowdown is set to yes, this indicates the maximum time for which a publisher will wait while trying to push the messages to the PSQ after it has reached 95% of its maximum limit. This applies to all publishers.

  Navigated to:

  FioranoMQ->fiorano->mq->pubsub->TopicSubSystem->MaxPublisherBlockTime

  The default value of this parameter is 120000 milliseconds.

Points to Note:

The Pubsub algorithm needs a new profile to run the server. The older profile has to be ported to the new profile. To port the older profile to the current profile, configurations need to be replicated manually. It server is recommended that the Server run with JVM heap size with the following setting to be made in fiorano_vars.bat/.sh:

JVM_SERVER_ARGS=-server -Xms256m -Xmx512m (-Xms512m -Xmx1024m for enterprise operation)

Non-persistent messages will always be dropped if PSQ size exceeds MaxPersistentStoreSize. However, non-persistent messages can persist till the Server goes out of disk space by setting its value to -1.

Flow control: Backoff algorithm will kick-in if PSQ size crosses PublishBackoffThreshold. By default, when 60% of the PSQ is used, every published call will be blocked for PublishWait (50ms by default). With every 5% increase in the PSQ size, publish call blocking time will be doubled. At the same time, the publish call blocking time will be halved with every 5% decrease. If 95% of the PSQ is used, the publish call will block calls for an infinite time. The publish call blocking time will reduce to 1600 ms if the PSQ size goes down to 85%.

The new back off algorithm can be disabled by setting EnablePublisherSlowdown to **No**.

The table below gives the default Blocking Time during various states of PSQ:

% of PSQ Used	Publish Block Time
60 %	50 ms (PublishWait default)
65 %	100 ms
70 %	200 ms
75 %	400 ms
80 %	800 ms
85 %	1600 ms
90 %	3200 ms
95 %	Block till PSQ size reduces to 85%

Calculating Memory Requirements for FioranoMQ Server

Approximate minimum memory allocated by default for each module in the FioranoMQ Server is given in the table below:

Module	Minimum Amount of Memory Required
Topic Connection	128 KB This is the default connection buffer size associated with each topic connection. This value is determined by the parameter BasicAllowedSize in the Topic Subsystem. More information about this parameter can be found in **Chapter 4: Topic Subsystem Level Configuration** [Link?] of the *FioranoMQ Reference Guide*. Please see Section 25.2.2 Setting in-memory buffers for Subscribers for how to change this parameter.
Topic Manager (Unique for all topics)	512 KB This is the default maximum in-memory size of the Message buffer that is used across all the storage topics in FioranoMQ. This value is determined by the parameter MaxBufferSize in the GMS Topic Manager. This can be set using Admin Studio in Offline editing mode by navigating to Fiorano > mq > pubsub > databases > file > TgmsManager > MaxBufferSize.
Queue	1024 KB (Non-persistent buffer size) This is the default buffer size for the Non-Persistent Buffer associated with each storage queue. This value is determined by the parameter InMemoryBufferSize in the Queue Subsystem/Queue. More information about this parameter can be found in **Chapter 3: Queue Subsystem Level Configurations** [Link?] of the *FioranoMQ Reference Guide*. Please see section 29.1 Support for Destination Level Configuration for referencing the difference between setting a parameter value at Subsystem level and at the Destination level. 512 KB (Persistent buffer size) This is the default buffer size for the Persistent Buffer associated with each storage queue. This value is determined by the parameter PersistentInMemoryBufferSize in the Queue Subsystem/Queue. More information about this parameter can be found in **Chapter 3: Queue Subsystem Level Configurations** of the *FioranoMQ Reference Guide*. Please see section 29.1 Support for Destination Level Configuration for referencing the difference between setting a parameter value at Subsystem level and at the Destination level.
Queue Receiver	256 KB This is the default maximum buffer size associated with each Queue Receiver when retrieving the messages from the Server DB. This value is determined by the parameter MaxPrefetchSize in the Queue Subsystem. More information about this parameter can be found in **Chapter 3: Queue Subsystem Level Configurations** of the *FioranoMQ Reference Guide*. Please see section 25.1.1.5 PTP Prefetch size to learn how to change this parameter and to learn of its

Module	Minimum Amount of Memory Required
	importance.
Route	128 KB This is the default buffer size associated with the Route subscriber. This parameter comes into use when a Route is created from a Topic (Source) to a Queue (Target). This value is determined by the parameter MaxTopicBuffer of the Route Manager. More information about this parameter can be found in **Chapter 15: Advance Configuration** [Link?] of the *FioranoMQ Reference Guide*.
Total	2560 KB

The minimum amount of memory consumed by the JMS objects inside FioranoMQ Server JVM can be calculated using the formula below:

Minimum memory required = (Number of topic connections * 128) + 512 + (Number of Queues * 1536) + (Number of Queue Receivers * 256) + (Number of Routes * 128) (In Kilobytes).

Note: These calculations are only done to determine the minimum amount of memory consumed by the JMS Objects created inside the FioranoMQ Server. This is not sufficient amount of memory to deploy the FioranoMQ Server. The amount of memory necessary to deploy the FioranoMQ Server is generally higher than the value obtained in these calculations.