

Background Execution

- [Overview](#)
- [Running Queue](#)
- [Waiting Queue](#)
- [Settings](#)
 - [Concurrent Tasks](#)
 - [Initial Wait Time](#)
 - [Maximum Execution Time](#)

Overview

[top](#)

This page is designed to allow you to manage the queue of report queries in real time.

The screenshot shows a window titled 'Background Execution'. On the left, there is a sidebar with instructions: 'Let's make this easy. You can manage all your report queries here, viewing and taking action on queries both queued and running. Queries can be cancelled or promoted. Need help? Visit the Yellowfin forum or wiki to find the answers you are looking for.' The main area has a 'Refresh' button and two sections: 'Running' and 'Waiting'. The 'Running' section contains a table with one row: 'Sales by Artist & Genre' requested by 'System Administrator', started on '10/8/2016 13:41:46', with a running time of '67.337s', and state 'Cross Tab'. The 'Waiting' section is empty, showing 'None Found'.

Running Queue

[top](#)

This list shows all queries that are currently running against their respective data sources.

Running					
Name	Requestor	Started Running	Running Time	State	
Artist Transactions	Mauve Jordan	12/8/2016 10:3:32	7.479s	Query	
Sales by Artist & Genre	Mauve Jordan	12/8/2016 10:3:32	7.482s	Query	
Sales by Artist & Genre	Violet Southgate	12/8/2016 10:2:58	41.046s	Cross Tab	
Store Sales	Mauve Jordan	12/8/2016 10:3:32	7.484s	Query	

The following information is available in the list:

Item	Description
Name	This displays the name of the report associated with the running query.
Requester	This displays the name of the user that is running the report.
Started Running	This displays the date and time the query started running (based on the Yellowfin server time, not user time).
Running Time	This displays the length of time the report query has been running.
State	This displays the state of the query. Possible values are: <ul style="list-style-type: none"> ▪ Initialising ▪ Query ▪ Cross Tab ▪ Chart
(Cancel)	This cancels the selected query and removes it from the queue.

Waiting Queue

[top](#)

This list shows all queries that are currently waiting to be run. Queries will progress to the **Running** queue when a report completes or times out.

Running					
Name	Requestor	Started Running	Running Time	State	
Artist Transactions	Mauve Jordan	12/8/2016 10:3:32	7.479s	Query	
Sales by Artist & Genre	Mauve Jordan	12/8/2016 10:3:32	7.482s	Query	
Sales by Artist & Genre	Violet Southgate	12/8/2016 10:2:58	41.046s	Cross Tab	
Store Sales	Mauve Jordan	12/8/2016 10:3:32	7.484s	Query	

The following information is available in the list:

Item	Description
Name	This displays the name of the report associated with the running query.
Requester	This displays the name of the user that is running the report.
Enqueued Time	This displays the date and time the user started running the report (based on the Yellowfin server time, not user time).
Waiting Time	This displays the length of time the report query has been in this queue.
 (Add)	This pushes the query out of the waiting queue and into the running queue, regardless of queue limit.
 (Cancel)	This cancels the selected query and removes it from the queue.

Settings

[top](#)

In order to control how the background execution queue works you will need to modify some parameters found in the web.xml.

Locate the following section in your web.xml file (found in `Yellowfin\appserver\webapps\ROOT\WEB-INF\`) and uncomment the parameters you need to adjust.

```

<!-- Execution queue values -->
<!-- Uncomment these values if you want to customise the behaviour of the background execution queue.

        ExecutionQueueConcurrentTasks:      The number of reports that can be run concurrently.
                                                Default: 5 times the number of
CPU cores available on the server.
        ExecutionQueueInitialWaitTime:      How long the server will wait (in seconds) to finish a report
before
                                                the client shows the delivery
popup and start polling.
                                                Default: 5 seconds
        ExecutionQueueMaximumExecutionTime: The maximum time (in minutes) the reports can be running in the
are killed.
                                                execution queue before they
                                                Default: 240 minutes (4 hours)
<init-param>
  <param-name>ExecutionQueueConcurrentTasks</param-name>
  <param-value>10</param-value>
</init-param>
<init-param>
  <param-name>ExecutionQueueInitialWaitTime</param-name>
  <param-value>5</param-value>
</init-param>
<init-param>
  <param-name>ExecutionQueueMaximumExecutionTime</param-name>
  <param-value>240</param-value>
</init-param>
-->

```

Concurrent Tasks

This option allows you to configure the number of reports that can be in the **Running** queue at the same time.

Default value: 5 x the number of CPU cores available on the server.

```

<init-param>
  <param-name>ExecutionQueueConcurrentTasks</param-name>
  <param-value>10</param-value>
</init-param>

```

Initial Wait Time

This option allows you to configure how long the server will wait (in seconds) to finish a report before the client shows the delivery popup and starts polling.

Default value: 5 seconds.

```

<init-param>
  <param-name>ExecutionQueueInitialWaitTime</param-name>
  <param-value>5</param-value>
</init-param>

```

Maximum Execution Time

This option allows you to configure the maximum time (in minutes) a report can be in the **Running** queue before it is stopped and removed.

Default value: 240 minutes (4 hours)

```
<init-param>  
  <param-name>ExecutionQueueMaximumExecutionTime</param-name>  
  <param-value>240</param-value>  
</init-param>
```

[top](#)