Queue Insights Documentation

Please contact the .NET Engineering Services Team with any questions about this service.

Queue Insights is a GitHub check designed to show developers the current status of Helix, and information about what infrastructure a PR uses.

It's important to note that Queue Insights data is specific to the context of the PR.

Was this helpful?

At the bottom of every Queue Insights check is a line of text stating *"Was this helpful? :thumbsup: :thumbsdown:"*. Selecting either thumbs up/down will direct you to our feedback page.

This feedback page **captures context** about where you clicked this button. We are able to see which PR this happened on, the time, etc. to help us diagnose and understand the event that caused your feedback.

If you decide to not remain anonymous, and leave your GitHub username, **We will follow-up with your feedback**. In addition, negative feedback creates a GitHub issue in dotnet/arcade that we will track.

Information Age

The information shown in Queue Insights is cached, updating daily, and consists of information over the past 30 days. Any queue or build machine used by your repo against a specific target branch will be shown for 30 days.

This means that if your repo stops using a given Helix queue, Queue Insights will show that queue for 30 days before it no longer appears in the check.

Pipeline Predictions

Here, we show the lower, expected, and higher estimated times for a pipeline the PR uses to complete, not including time spent in queue waiting for a build machine.

You should interpret this data as this pipeline should finish around the *expected* time but may finish as quick as the *lower* time or take as long as the *higher* time.

This data is calculated from our build telemetry, and you may visit our one-pager for how it works.

Multi-modal distributions

Some pipelines we cannot give an accurate prediction for. Typically, this is because the pipeline changes what stages run depending on the contents of the PR. Since we cannot account for this yet, you will experience an :x: symbol in the place of a prediction.

Known Issues

If a Critical Known Issue has been created, we will not show a prediction, and instead inform you that there is a critical known issue that could affect the estimated times.

Not enough data

If your pipeline is brand new or changed definition id's for instance, we won't have enough data to create predictions for. Currently, we require **30** runs of your pipeline before we can create an accurate enough model.

Highest Work Item Wait Times

In this section, we show the top **5** (or less, depending on how many queues your PR uses) *work item wait times*. A work item is the smallest unit of work a queue can process in Helix. A Helix machine that processes work only runs one work item at a time.

Work item wait time is defined as the time a work item spends waiting to run. This is the amount of time the work item spends in queue waiting to run. A high work item wait time means that there are a lot of work items in that queue, and thus it might take longer to process work for that queue.

Moving Average

In this column, we show the moving average of the work item wait time, over the period of **24 hours**. This can be used to gauge if there is a sudden high influx of work items to a specific queue, which should be investigated with our *Grafana dashboard*

Grafana dashboard

Each queue mentioned in the Queue Insights check is hyperlinked, and will direct you to our Grafana dashboard **for that specific queue**. Here, you can dive deep into our infrastructure and view data like:

- Queue depth (the amount of work items waiting in that queue)
- Work item wait time
- Work items queued by repository

- Machine status (Busy processing work, provisioning, spinning down)

Queue List

In this section, we show you a list of queues that this PR will use. **This is filtered by the queues the PR uses, not the overall repository.**. This is further separated by two factors, *on-premises* queues and *Azure queues*.

:cloud: Azure-Hosted Queues

This list shows queues that are supported by Azure VMs. This means that the queue is powered by an Azure Virtual Machine Scale Set, meaning that we have the capacity to scale the work items the queue can process up and down depending on our budget and work items queued.

:office: Off-Prem (build lab) Queues

This list are queues that are hosted in a data center managed by DDFUN. These queues have a fixed capacity and are typically always on, so there is the added benefit of no wait time to provision a new VM, with the issue of not being able to scale these machines.

Build Pools

This section shows you what build pools are used, separated by 1ES Hosted and Microsoft Hosted. There are hyperlinks here to direct you to documentation about packages and software installed by default on these build machines.

Was this helpful? 👍 👎