

Predicting Input Queuing Time via Expected Queueing Time

tdresser@chromium.org

For the [Unresponsive Main Thread Scrolling Intervention](#), we need to determine when the main thread is unresponsive. We've proposed doing this using the Expected Queueing Time (EQT) metric; however, we've never quantified how well past expected queueing time predicts current queueing time.

The existing metrics in this space are:

RendererScheduler.ExpectedTaskQueueingDuration (EQT)

RendererScheduler.ImmediateTaskQueueingDuration (Actual Queueing Time)

There are two primary problems with measuring EQT's usefulness for predicting queueing time via `RendererScheduler.ImmediateTaskQueueingDuration`:

- The EQT UMA is reported once per second, and real events aren't distributed uniformly.
- We'd like a user input specific metric.

New Proposal

I propose we add a few metrics indicating how well this metric performs when trying to guess if input will be queued for longer than some threshold:

These metrics will be named:

- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeGreaterThan10ms`
- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeLessThan10ms`
- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeGreaterThan150ms`
- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeLessThan150ms`
- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeGreaterThan300ms`
- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeLessThan300ms`
- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeGreaterThan450ms`
- `RendererScheduler.QueueingDurationWhenExpectedQueueingTimeLessThan450ms`

We hope to see that when $EQT < \text{the threshold}$, QT is $< \text{the threshold}$ 80% of the time, and when $EQT > \text{the threshold}$, QT is $> \text{the threshold}$ > 95% of the time.

Old Proposal

`RendererScheduler.ExpectedToActualInputTaskQueueingDurationRatio`, which reports, per input event, the ratio between the EQT at the time the event was queued, to the actual queueing time.

We hope to see this metric log normally distributed around 1, with minimal variance. A mean other than 1 would indicate that we're consistently over or under predicting the actual queueing time, and high standard deviation would mean that we aren't predicting actual queueing time with any certainty.