

HdrHistogram

A High Dynamic Range (HDR) Histogram

View the [JavaDoc](http://www.javadoc.io/doc/org.hdrhistogram/HdrHistogram) www.javadoc.io/doc/org.hdrhistogram/HdrHistogram Find on [Maven Central](https://mvnrepository.com/artifact/org.hdrhistogram) [org.hdrhistogram](https://mvnrepository.com/artifact/org.hdrhistogram)

View implementations on GitHub:

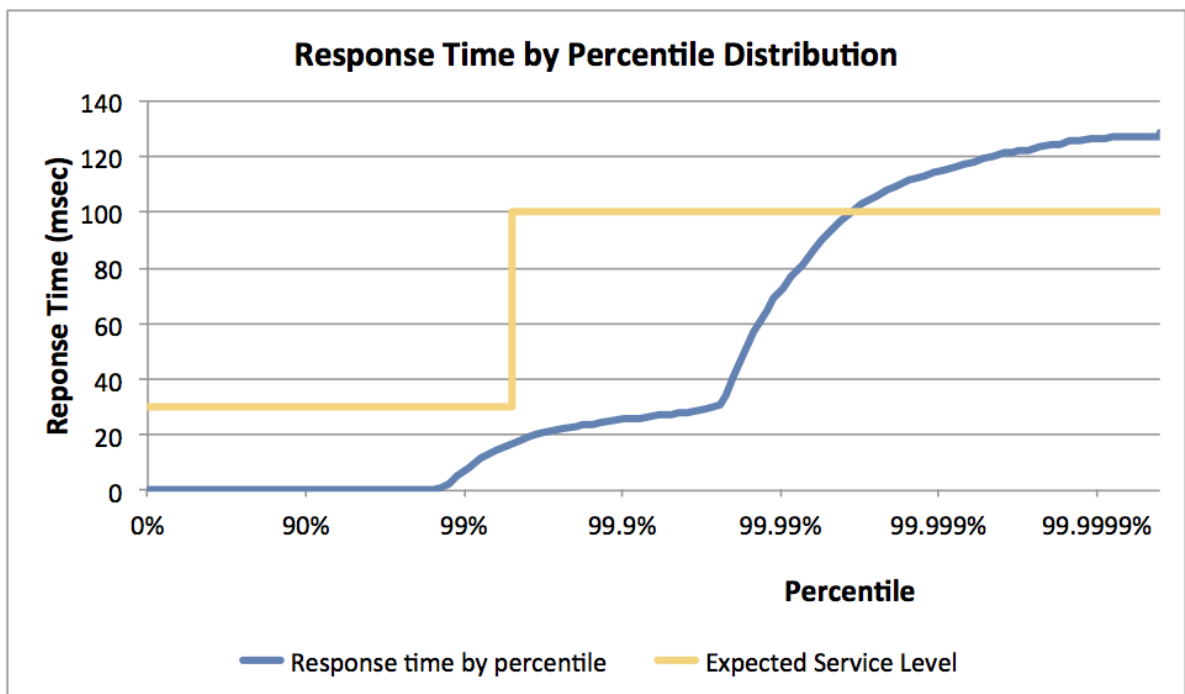
[Java](#) [HdrHistogram/HdrHistogram](#) [JavaScript](#) [HdrHistogram/HdrHistogramJS](#) [C](#) [HdrHistogram/HdrHistogram_c](#)

[C#/.NET](#) [HdrHistogram/HdrHistogram.NET](#) [Python](#) [HdrHistogram/HdrHistogram_py](#)

[Erlang](#) [HdrHistogram/hdr_histogram_erl](#) [Rust](#) [HdrHistogram/HdrHistogram_rust](#) [Go](#) [HdrHistogram/hdrhistogram-go](#)

Plot histogram file(s)

An example plot of an HdrHistogram based full percentile spectrum plot:



HdrHistogram: A High Dynamic Range Histogram.

A Histogram that supports recording and analyzing sampled data value counts across a configurable integer value range with configurable value precision within the range. Value precision is expressed as the number of significant digits in the value recording, and provides control over value quantization behavior across the value range and the subsequent value resolution at any given level.

For example, a [Histogram](#) could be configured to track the counts of observed integer values between 0 and 3,600,000,000,000 while maintaining a value precision of 3 significant digits across that range. Value quantization within the range will thus be no larger than 1/1,000th (or 0.1%) of any value. This example Histogram could be used to track and analyze the counts of observed response times ranging between 1 nanosecond and 1 hour in magnitude, while maintaining a value resolution of 1 nanosecond up to 1 microsecond, a resolution of 1 microsecond up to 1 millisecond, a resolution of 1 millisecond (or better) up to one second, and a resolution of 1 second (or better) up to 1,000 seconds. At it's maximum tracked value (1 hour), it would still maintain a resolution of 3.6 seconds.

HDR Histogram is designed for recoding histograms of value measurements in latency and performance sensitive applications. Measurements show value recording times as low as 3-6 nanoseconds on modern (circa 2014) Intel CPUs. The HDR Histogram maintains a fixed cost in both space and time. A Histogram's memory footprint is constant, with no allocation operations involved in recording data values or in iterating through them. The memory footprint is fixed regardless of the number of data value samples recorded, and depends solely on the dynamic range and precision chosen. The amount of work involved in recording a sample is constant, and directly computes storage index locations such that no iteration or searching is ever involved in recording data values.

Authors, Contributors, and License

HdrHistogram was originally authored by Gil Tene ([@giltene](#)) and placed in the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>, with ports by Mike Barker ([@mikeb01](#)) (C), Matt Warren ([@mattwarren](#)) (C#), Darach Ennis ([@darach](#)) (Erlang), Alec Hothan ([@ahothan](#)) (Python), Coda Hale ([@codahale](#)) (Go), Jon Gjengset ([@jonhoo](#)) (Rust), and Alexandre Victoor ([@alexvictoor](#)) (JavaScript)

Support or Contact

Don't call me, I won't call you.
