

Why is there a large gap between some of the Windows system metrics indices?

devblogs.microsoft.com/oldnewthing/20230410-00

April 10, 2023



Raymond Chen

The `GetSystemMetrics` function lets you query a grab bag of system metrics, primarily related to user interface matters. You pass in the magic number representing a metric, and the function returns the value of that metric.

Historically, the magic numbers for these metrics were small integers, starting with `SM_CXSCREEN` at zero, and continuing sequentially.

And then Windows 2000 introduced `SM_REMOTESESSION` with the value way out at `0x1000`. Why such a large value?

That particular gap was an artifact of the internal organization of the Windows team. The Remote Session metric is not really a user interface metric, even though the `GetSystemMetrics` function is in the user interface library. The Terminal Services folks kind of jammed that one into the `GetSystemMetrics` function, but since they knew they were treading on another team's turf, they put their index way out in a different part of the index space so they wouldn't mess up the user interface team's system of having all the metrics just be indices into an internal array of `int` values.

```
int GetSystemMetrics(int index)
{
    if (index == SM_REMOTESESSION) {
        return AskTerminalServicesIfSessionIsRemote();
    }

    /* rest of code unchanged */
    if (index >= 0 && index < SM_CMETRICS) {
        return metricsTable[index];
    }

    SetLastError(ERROR_INVALID_PARAMETER);
    return 0;
}
```

Next up is Windows XP, which introduced `SM_SHUTTINGDOWN` with the value way out at `0x2000`. The user interface team decided that they would keep the low-index metrics for true metrics (things that represent the size or number of other things), and any new metrics that are really just Boolean values disguised as metrics should be placed in the range starting at `0x2000`.

That's why you have new metrics like `SM_SHUTTINGDOWN` and `SM_CARETBLINKING-ENABLED` out in the `0x2000` region.

This change roughly parallels the change in the assignment of `SystemParametersInfo` parameter indices: Historically, system parameter indices were also small integers, but at some point the user interface team decided to organize them a bit, and put all the Boolean-valued parameters in the `0x1000` range, and all of the integer-valued parameters go in the `0x2000` range. This simplified parameter validation since all of the integer or Boolean parameters can be validated in common code.

For system metrics, however, this doesn't really provide that much of a simplification. You could still have put the values in the array of integers. There's no validation necessary beyond the index itself, unlike `SystemParametersInfo` which had to validate pointers and structure sizes.

I mean, sure, you saved a tiny bit of memory by packing a bunch of Boolean metrics into a bitmask, but that's a meager savings of just a few dozen bytes at most. The code necessary to manage the bitmask is probably already bigger than the data savings.

Well, what's done is done. And that's why there are gaps in the system metrics index values.