



# Intel® SoC Watch for Windows\*

## Release Notes

October 2020

Version 2020.4

Intel Corporation

[www.intel.com](http://www.intel.com)

[Legal Information](#)

---

# Contents

<b>Legal Information .....</b>	<b>3</b>
<b>Version History .....</b>	<b>4</b>
<b>Customer Support.....</b>	<b>5</b>
<b>Chapter 1: Introduction</b>	
<b>Chapter 2: New in This Release</b>	
<b>Chapter 3: System Requirements</b>	
Minimum System Requirements .....	8
<b>Chapter 4: Where to Find the Release</b>	
<b>Chapter 5: Installation Notes</b>	
<b>Chapter 6: Fixed Issues</b>	
<b>Chapter 7: Known Issues</b>	
<b>Chapter 8: Related Documentation</b>	
<b>Chapter 9: Acronyms and Terms</b>	

# Legal Information

---

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel, the Intel logo, Intel Atom, Intel Core, Intel Xeon Phi, VTune and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

Microsoft, Windows, and the Windows logo are trademarks, or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Java is a registered trademark of Oracle and/or its affiliates.

OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos.

© **Intel Corporation**

---

# Version History

---

These are the main releases of Intel® SoC Watch:

<b>Date</b>	<b>Revision</b>	<b>Description</b>
June, 2019	2.11	Improves handling of unrecognized CPUs, reporting S-state when hibernation occurs, and other bug fixes.
September, 2019	2019.12	Added support for Intel platform code named Ice Lake. Modified hw-cpu-pstate reporting.
October, 2019	2019.13	Fixed issue in hw-cpu-pstate for Intel platform code named Ice Lake.
November, 2019	2020.1	Added support for Intel platform code named Comet Lake.
February, 2020	2020.2	Added collection of tool usage analytics. Added new features pch-slps0, pch-slps0-dbg. Improved error messages and help output. Enhanced driver security.
June, 2020	2020.3	Bug fix release.
July, 2020	2020.3.1	Bug fixes
September, 2020	2020.3.2	Bug fixes including correcting hibernation detection.
October, 2020	2020.4	Added support for Intel platform code named Tiger Lake. Added topology label in reports for some metrics. Re-named feature cpu-gpu-concurrency to cpu-igpu-concurrency. Removed support for older platforms.

# *Customer Support*

---

For technical support, including answers to questions not addressed in this product, Intel® oneAPI IoT Toolkit forum ( <https://community.intel.com/t5/Intel-oneAPI-IoT-Toolkit/bd-p/oneapi-iot-toolkit>).

# Introduction

Intel® SoC Watch is a data collector for power-related data that can help identify issues on a platform that prevent entry to power-saving states. Captured metrics include:

- System sleep states
- CPU and GPU sleep states
- Processor frequencies
- Temperature data
- Device sleep states

You can correlate the collected data and visualize over time using Intel®VTune Profiler.

This document provides system requirements, installation instructions, issues and limitations, and legal information.

To learn more about this product, see:

- New features listed in the [New in This Release](#) section below, or in the help.
- Reference documentation listed in the [Related Documentation](#) section below
- Installation instructions can be found in the [Installation Notes](#) section below.
- For a detailed quick start guide to running the tool, see the *Intel SoC Watch User's Guide* in your installed documentation.

## Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

## 2

## New in This Release

The 2020.4 release (driver v2.14.0) contains these changes:

- Intel® SoC Watch reports now include additional topology information in column headers and row entries for some features. A change was required to differentiate data reported for multiple discrete graphics cards of the same type and discrete graphics card with multiple tiles. This change extended into the reporting of CPU and integrated graphics metrics due to the common underlying tool framework. The topology components are separated by "/" and the number of components in the topology string will vary depending on the topology of the target platform. For example, you will see "CPU/Package\_0/Core\_0" shown where in prior releases it was simply "Core\_0". For integrated graphics, you now see "iGPU/Graphics" rather than "Graphics". For discrete graphics, you see "dGPU/DG1\_173:0:0/Tile\_0" where the Bus:Device:Function for the card is appended to uniquely identify its data. Reports for metrics that are system level or from PCH are unchanged.
- Changed feature name `-f cpu-gpu-concurrency` to `-f cpu-igpu-concurrency` to make it clear this refers to integrated GPU vs. discrete GPU. The table name was changed accordingly to "CPU-iGPU Concurrency Summary".
- Changed table names for features `-f hw-igfx-cstate` and `-f hw-igfx-pstate` to include "Integrated". Previously, these output tables did not identify the results as Integrated Graphics vs. Discrete Graphics.
- Added support for Intel platform code named Tiger Lake.
- Added note to the report for feature `-f os-cpu-pstate` to explain the appearance of -1 as a frequency value. When no P-state change event occurs for one or more threads, the following message is written at the start of the report: "Note: No OS P-state event is generated for a thread if it does not change frequency during the collection period. Frequency is reported as -1 when this occurs."
- Support for the following older platforms has been removed in this release: Intel platforms code named Anniedale, Valleyview, Cherry Trail, Haswell, and Broadwell.

# *System Requirements*

---

## **Minimum System Requirements**

---

You can run Intel SoC Watch on these 64-bit operating systems with administrator permissions:

- Windows\* 10

# *Where to Find the Release*



Intel SoC Watch is available in Intel® oneAPI Base Toolkit as a part of Intel® VTune™ Profiler.

# Installation Notes

Intel SoC Watch for Windows\* OS is installed as part of the Intel® VTune™ Profiler, which is included in the Intel® oneAPI Base Toolkit. Intel SoC Watch is used by Intel® VTune™ Profiler for collecting certain analysis, but it is also a standalone tool. Following are instructions for setting up Intel SoC Watch for standalone use.

## **When Intel® oneAPI Base Toolkit is used to install Intel® VTune™ Profiler and it is installed directly on the target system:**

Intel SoC Watch will already be installed and is located in `C:\Program Files (x86)\Intel\oneAPI\vtune\<oneAPI-release-version>\socwatch`.

Configure the Intel SoC Watch collection environment for standalone use by opening a command window and executing the following command:

```
set PATH="%Program Files (x86)\Intel\oneAPI\vtune\<oneAPI-release-version>\socwatch\64";%PATH%
```

The Intel SoC Watch release notes and user's guides are located in `C:\Program Files (x86)\Intel\oneAPI\vtune\<oneAPI-release-version>\documentation\en\socwatch`.

## **When Intel® oneAPI Base Toolkit is used to install Intel® VTune™ Profiler and it is installed on a host system:**

Intel SoC Watch will need to be copied to the target system and installed. The install package is located on the host system at `C:\Program Files (x86)\Intel\oneAPI\vtune\<oneAPI-release-version>\target\windows\vtune_profiler_target_x86_64.zip`. Copy `vtune_profiler_target_x86_64.zip` to the target system and extract it into a directory of your choice `<VTUNE_DIR>`.

Configure the Intel SoC Watch collection environment for standalone use by opening a command window and executing the following command:

```
set PATH=<VTUNE_DIR>\socwatch\64;%PATH%
```

The release notes and user's guide are not installed with this package. They are located on the host system in `C:\Program Files (x86)\Intel\oneAPI\vtune\<oneAPI-release-version>\documentation\en\socwatch`.

## **Installing WDTF to Enable --auto-connected-standby**

Use of the `--auto-connected-standby` option requires the Windows\* OS Driver Test Framework (WDTF) to be installed on the target system. WDTF is found in the Windows Driver Kit (WDK). Below are instructions.

1. Get the latest WDK from Microsoft <https://docs.microsoft.com/en-us/windows-hardware/drivers/download-the-wdk>. Follow only the instructions for installing the WDK.
2. After installing the WDK, open an administrative command prompt in this folder: `C:\Program Files (x86)\Windows Kits\10\Testing\Runtimes`
3. Enter the following command to install WDTF:

```
msiexec /i "Windows Driver Testing Framework (WDTF) Runtime Libraries-x64_en-us.msi" /1* WDTFInstall.log WDTFDir=c:\wdtf WDTF_SKIP_MACHINE_CONFIG=1
```

## Fixed Issues

---



The 2020.4 release has a fix for these issues.

- Re-enabled features -f acpi-sstate, -f pch-slps0-dbg, -f pkg-pwr, and -f dram-pwr.
- Use of feature -f ia-throt-rsn on Intel platform code named Tiger Lake is disabled when Windows\* OS Hypervisor Enforced Code Integrity (HECI) is enabled. Enabling this security mode results in some register access being blocked unless allowed by the Windows\* OS.

# Known Issues

## Bandwidth (on Intel Core Platforms)

- The presence of EDRAM on a system may not be detected by Intel SoC Watch. This is known to occur when the accelerator card VCA2, which contains EDRAM, is present.
- Total DDR bandwidth does not include EDRAM. On systems using EDRAM, the `ddr-bw` feature report may have a discrepancy between the total data reads and writes and the total component requests. The Data Reads+Data Writes will be significantly higher than the total IA+GT+IO requests, because the EDRAM requests are not included.

## Bandwidth and DRAM Self Refresh (on Intel Atom® Platforms)

- On Intel platforms code named Apollo Lake and Gemini Lake, memory bandwidth and memory self-refresh metrics are not available. These features are not supported: `ddr-bw`, `cpu-ddr-bw`, `cpu-ddr-mod0-bw`, `cpu-ddr-mod1-bw`, `disp-ddr-bw`, `isp-ddr-bw`, `gfx-ddr-bw`, `io-bw`, `all-approx-bw`, `dram-srr`.

## C-States / P-States

- When collecting a trace of residency data from hardware counters (i.e., using `-m`), the summarized residency data could be 2-3% inaccurate due to error propagation in the accumulation of each sample's calculated residency. Collecting without `-m` results in greater accuracy because only a single sample is taken. However, long collection duration could result in a counter rollover, and that will not be detected without the use of `-m`.
- The hardware CPU P-state data may be missing for some Cores when using feature `-f hw-cpu-pstate` on Intel platforms code named Skylake, Kaby Lake, Whiskey Lake, and Amber Lake. The issue is caused by unexpected behavior of the hardware counters. The tool ignores these bad samples which results in the missing data.
- On Intel platforms code named Broxton and Apollo Lake, the `cpu-cstate` metric results do not contain module C-state information.
- OS-based CPU P-state report does not support platforms with > 64 logical processors. The hardware-based P-state report is correct. The problem is in the OS event trace which provides state changes for only a partial set of logical processors when the platform is configured for more than 64 logical processors. When this situation occurs, a warning message is printed in the report indicating which processors have no OS P-state change events.

## S States & D States

- Occasional missing reports for `acpi-dstate`, `os-cpu-cstate` and other OS/ACPI metrics which are based on OS event trace data. This issue occurs when there are insufficient OS event trace sessions available to Intel SoC Watch when a collection was started. SoC Watch v2020.3.2 displays the following error messages when this issue is detected: *Warning: Cannot enable provider in the trace file XXXX\_extraSession.etl. Insufficient system resources exist to complete the requested service. Warning: Failed to start EtICollector.* To resolve the problem, Event Trace Sessions must be made available either by trying again to see if other processes have closed their sessions, or running Windows Performance Monitor to view the Data Collector Sets/Event Trace Sessions that are running and stopping one of them.
- Some disk devices are being excluded from the `-f acpi-dstate` report when using the Intel SoC Watch for Windows\* OneCore package. To workaround the issue, remove the filter by adding `--option no-device-filter` to your command line. The issue occurs because the Device Class information is not available causing a gap in the filter. An alternative is under investigation.

## Miscellaneous

- On ChromeOS, some SoCWatch collections may delay their start until after a waking up from S0iX sleep under certain conditions. There are two known alternative solutions: 1) Unload the socwatch.ko kernel module, allowing socwatch to run in "driverless" mode. 2) Avoid the use of wrapper scripts to put the system into S0ix and instead use SoCWatch's built-in capability for this (the "-z" feature switch). This will force the system to sleep for the duration of the "-t" switch, and relevant outputs will reflect the time spent in sleep.
- If collecting a large number of metrics and requesting multiple types of results files to be generated on the same command line, SoC Watch may report the following, *Warning: Could not post process metric data: Too many open file handles. Results may be incomplete for some metrics. Try post processing results with only one -r option at a time. If the problem persists; try collecting fewer metrics.* There will be some missing reports in the results files that are generated if this occurs. The work around is to specify only one -r file type at collection time or collect fewer metrics. After collection, use -i option with each of the remaining file types to generate the additional result files.
- When using -z ( --auto-connected-standby ), the following failed message is displayed by WDTF, but can be ignored when followed by the enter /exit messages from WDTF as shown below:

```
Loading Wex.Logger.dll from systemdata.cpp line# 922 failed.
```

```
WDTF_SYSTEM : INFO : Attempt Entering Connected Standby At (hh:mm:ss): 13:4:35 ,  
Wake Time In Milliseconds: 20000
```

```
WDTF_SYSTEM : INFO : Exiting Connected Standby. Elapsed time (hh:mm:ss): 0:0:20
```

- Feature -f dram-pwr is not supported by all versions of the server Intel platforms code named Skylake-Xeon, Cascade Lake-Xeon, and Denverton). The report contains all zero values in this case.
- On platforms with HyperV enabled, the sum of the CPU C-state idle residencies will not match CPU P-state idle residency time. The issue occurs because HyperV does not allow setting of a bit, resulting in incorrect core-level reporting of both C-state and P-states.
- Metrics report Unknown 0 when -m is not used and hibernation occurs. Metrics with a snapshot default collection mode, such as CPU C-state, will show the Unknown state with 0 time and the remaining states will not sum to the total collection duration if the system entered hibernation during the collection and the -m option was not specified. The snapshot metrics are only collected at the start and end of a collection by default, but finding hibernation time requires samples taken throughout the collection. Including -m will cause continuous sampling to occur for all metrics. When hibernation occurs, a message reporting time spent in hibernation appears at the beginning of the summary report. The Unknown state is then included for all appropriate metrics and the time in hibernation is included in that state. Refer to the *Intel SoC Watch User's Guide* "Options Quick Reference" section to learn which metrics have a snapshot collection mode by default.
- Package level power data (-f pkg-pwr) is reported incorrectly for Cascade Lake-Xeon (AP) which has multiple Die in the CPU package. Intel SoC Watch labels the power as per package but it is actually per Die. There is no package level power.
- Syntax errors in the command line may not report a visible error message. If a collection did not run and you are not seeing any error message, add option -d 2 to your command line to get more information.
- Insufficient system resources error seen on occasion when collecting OS event trace metrics such as acpi-dstate. The system error "WARNING: Cannot enable provider in the trace file <etl filename>" has been reported when collecting metrics that enable event trace logging. This error prevents ETL logging from being started and is usually caused by a background process consuming system resources. Use Task Manager to find and remove such processes then try the collection again.
- Hyper-V and Virtualization-based Security (VBS) prevent some metrics from being collected. Intel SoC Watch detects when Hyper-V and Virtualization-based Security is enabled on the platform, reports a warning message on the console and disables metrics that are blocked by these settings. When Hyper-V and VBS are enabled `cpu-gpu-concurrency` cannot be collected.
- Collection on Windows\* Server 2016 OS will result in failure to load the Intel SoC Watch driver if Secure Boot is enabled on the platform. The workaround is to disable Secure Boot. The problem on Windows 10 client OS was resolved.

- If a command window is closed (using either the X button or Alt-F4) while the socwatch process is running, or the Task Manager is used to kill the socwatch process, then the behavior of a subsequent run of Intel SoC Watch becomes unknown. The proper way to terminate Intel SoC Watch is using Ctrl-C. A collection driver may be left in an undefined state when Intel SoC Watch is abruptly terminated because there is no OS event to allow proper cleanup. This can cause the next Intel SoC Watch collection to result in anything from bad data to a system crash. If a driver is left running, it must be removed. You can reboot the system to clear a driver or use the following set of commands to check if the driver is running, stop it, and then delete it:  

```
sc query socwatchdrv; sc stop socwatchdrv; sc delete socwatchdrv.
```

### Intel® VTune™ Profiler Visualization

- When using Intel® VTune™ Profiler to view ddr-bw data collected by SoC Watch, the VTune summary report may not match the SoC Watch summary report. If this occurs, the SoC Watch summary results are correct. This issue has been seen on platforms prior to Intel platform code named Tiger Lake.
- The Intel VTune Profiler System Summary does not report the rated frequency for the CPU when viewing results from data collected by SoC Watch, such as Throttling Analysis. Instead, it reports 1GHz which is the clock frequency used in the calculations for processing the SoC Watch data.
- Intel VTune Amplifier 2017 for Systems Update 1 or later is required for visualizing and analyzing Intel SoC Watch v2.10.0 and newer PWR files. We recommend using the latest version of Intel VTune Profiler.
- If the bandwidth is 0 Mb throughout the collection for a particular bandwidth type, Intel VTune Profiler will not show a timeline entry for it. The timeline is shown only if there is at least one non-zero value.
- In some cases, the summary CSV results produced by Intel SoC Watch can vary from the summary results shown by Intel VTune Profiler even though they represent the same collection. For example, the summary CSV file may report a specific cpu-pstate residency of 50.78% and Intel VTune Profiler may report the same cpu-pstate residency as 50.8%.
- In order to visualize graphics C-states that are reported as Render and Media, the table headers in the trace file (generated with option -r int), must be manually modified, adding *Render* and *Media* to the appropriate C0, C1, and C6 column headers.

## ***Related Documentation***

---



The release contains these documents:

- Intel® SoC Watch for Windows OS\* User's Guide.
- Energy Analysis help (<https://software.intel.com/en-us/energy-analysis-user-guide>)

## *Acronyms and Terms*

The following acronyms and terms are used in this document (arranged in alphabetic order):

<b>Acronym/Term</b>	<b>Description</b>
SoC	System on Chip