Microsoft Corporation Technical Documentation License Agreement (Standard)

READ THIS! THIS IS A LEGAL AGREEMENT BETWEEN MICROSOFT CORPORATION ("MICROSOFT") AND THE RECIPIENT OF THESE MATERIALS, WHETHER AN INDIVIDUAL OR AN ENTITY ("YOU"). IF YOU HAVE ACCESSED THIS AGREEMENT IN THE PROCESS OF DOWNLOADING MATERIALS ("MATERIALS") FROM A MICROSOFT WEB SITE, BY CLICKING "I ACCEPT", DOWNLOADING, USING OR PROVIDING FEEDBACK ON THE MATERIALS, YOU AGREE TO THESE TERMS. IF THIS AGREEMENT IS ATTACHED TO MATERIALS, BY ACCESSING, USING OR PROVIDING FEEDBACK ON THE ATTACHED MATERIALS, YOU AGREE TO THESE TERMS.

1. For good and valuable consideration, the receipt and sufficiency of which are acknowledged, You and Microsoft agree as follows:

(a) If You are an authorized representative of the corporation or other entity designated below ("**Company**"), and such Company has executed a Microsoft Corporation Non-Disclosure Agreement that is not limited to a specific subject matter or event ("**Microsoft NDA**"), You represent that You have authority to act on behalf of Company and agree that the Confidential Information, as defined in the Microsoft NDA, is subject to the terms and conditions of the Microsoft NDA and that Company will treat the Confidential Information accordingly;

(b) If You are an individual, and have executed a Microsoft NDA, You agree that the Confidential Information, as defined in the Microsoft NDA, is subject to the terms and conditions of the Microsoft NDA and that You will treat the Confidential Information accordingly; or

(c)If a Microsoft NDA has not been executed, You (if You are an individual), or Company (if You are an authorized representative of Company), as applicable, agrees: (a) to refrain from disclosing or distributing the Confidential Information to any third party for five (5) years from the date of disclosure of the Confidential Information by Microsoft to Company/You; (b) to refrain from reproducing or summarizing the Confidential Information; and (c) to take reasonable security precautions, at least as great as the precautions it takes to protect its own confidential information, but no less than reasonable care, to keep confidential the Confidential Information. You/Company, however, may disclose Confidential Information in accordance with a judicial or other governmental order, provided You/Company either (i) gives Microsoft reasonable notice prior to such disclosure and to allow Microsoft a reasonable opportunity to seek a protective order or equivalent, or (ii) obtains written assurance from the applicable judicial or governmental entity that it will afford the Confidential Information the highest level of protection afforded under applicable law or regulation. Confidential Information shall not include any information, however designated, that: (i) is or subsequently becomes publicly available without Your/Company's breach of any obligation owed to Microsoft; (ii) became known to You/Company prior to Microsoft's disclosure of such information to You/Company pursuant to the terms of this Agreement; (iii) became known to You/Company from a source other than Microsoft other than by the breach of an obligation of confidentiality owed to Microsoft; or (iv) is independently developed by You/Company. For purposes of this paragraph, "Confidential Information" means nonpublic information that Microsoft designates as being confidential or which, under the circumstances surrounding disclosure ought to be treated as confidential by Recipient. "Confidential Information" includes, without limitation, information in tangible or intangible form relating to and/or including released or unreleased Microsoft software or hardware products, the marketing or promotion of any Microsoft product, Microsoft's business policies or practices, and information received from others that Microsoft is obligated to treat as confidential.

2. You may review these Materials only (a) as a reference to assist You in planning and designing Your product, service or technology ("Product") to interface with a Microsoft Product as described in these Materials; and (b) to provide feedback on these Materials to Microsoft. All other rights are retained by Microsoft; this agreement does not give You rights under any Microsoft patents. You may not (i) duplicate any part of these Materials, (ii) remove this agreement or any notices from these Materials, or (iii) give any part of these Materials, or assign or otherwise provide Your rights under this agreement, to anyone else.

3. These Materials may contain preliminary information or inaccuracies, and may not correctly represent any associated Microsoft Product as commercially released. All Materials are provided entirely "AS IS." To the extent permitted by law, MICROSOFT MAKES NO WARRANTY OF ANY KIND, DISCLAIMS ALL EXPRESS, IMPLIED AND STATUTORY WARRANTIES, AND ASSUMES NO LIABILITY TO YOU FOR ANY DAMAGES OF ANY TYPE IN CONNECTION WITH THESE MATERIALS OR ANY INTELLECTUAL PROPERTY IN THEM.

4. If You are an entity and (a) merge into another entity or (b) a controlling ownership interest in You changes, Your right to use these Materials automatically terminates and You must destroy them.

5. You have no obligation to give Microsoft any suggestions, comments or other feedback ("Feedback") relating to these Materials. However, any Feedback you voluntarily provide may be used in Microsoft Products and related specifications or other documentation (collectively, "Microsoft Offerings") which in turn may be relied upon by other third parties to develop their own Products. Accordingly, if You do give Microsoft Feedback on any version of these Materials or the Microsoft Offerings to which they apply, You agree: (a) Microsoft may freely use, reproduce, license, distribute, and otherwise commercialize Your Feedback in any Microsoft Offering; (b) You also grant third parties, without charge, only those patent rights necessary to enable other Products to use or interface with any specific parts of a Microsoft Product that incorporate Your Feedback; and (c) You will not give Microsoft any Feedback (i) that You have reason to believe is subject to any patent, copyright or other intellectual property claim or right of any third party; or (ii) subject to license terms which seek to require any Microsoft Offering incorporating or derived from such Feedback, or other Microsoft intellectual property, to be licensed to or otherwise shared with any third party.

6. Microsoft has no obligation to maintain confidentiality of any Microsoft Offering, but otherwise the confidentiality of Your Feedback, including Your identity as the source of such Feedback, is governed by Your NDA.

7. This agreement is governed by the laws of the State of Washington. Any dispute involving it must be brought in the federal or state superior courts located in King County, Washington, and You waive any defenses allowing the dispute to be litigated elsewhere. If there is litigation, the losing party must pay the other party's reasonable attorneys' fees, costs and other expenses. If any part of this agreement is unenforceable, it will be considered modified to the extent necessary to make it enforceable, and the remainder shall continue in effect. This agreement is the entire agreement between You and Microsoft concerning these Materials; it may be changed only by a written document signed by both You and Microsoft.

Notice

This document supports the preliminary and partial parts of the software program named Microsoft® Amalga™ distributed at the Software Design Review June 27 through June 29, 2011.

Date of Manufacture: June 2011

Microsoft Confidential. Can be used only pursuant to the Microsoft Corporation Technical Documentation License Agreement.

Investigating Connected Standby using Windows Performance Analyzer (WPA)

For connected standby feedback and questions, contact csfeedback@microsoft.com.
2/8/2013

Introduction

We rely on traces that are captured when the system is in Connected Standby (CS) in order to diagnose and fix problems with power consumption and battery life.

This document describes how to analyze traces for CS issues using the Windows Performance Analyzer tool (WPA). A step-by-step guide for analysis and overview of key CS concepts are provided.

Getting Started

This section covers steps to begin trace analysis with WPA and a brief description of key CS operational concepts.

Prerequisites

This guide assumes you have the following tasks completed:

- WPA already installed on your machine. The downloaded file that contains this documentation includes the installer for the WPA tool. See the Readme file for installation instructions.
- A CS trace already captured and have the trace (.etl) file ready. See below for instructions on how to capture a CS trace:
    1. Install WPT on the system in question. See the Readme file for installation instructions.
    2. Copy *trace_start.cmd* and *trace_end.cmd* to the system in question.
    3. Run *trace_start.cmd* from an elevated command prompt in the directory containing xperf. Then press the power button to put the machine into Connected Standby. Wait for at least 1 hour.
    4. Press the power button to wake up the machine. Then run *trace_end.cmd* from an elevated command prompt in the directory containing xperf to finish recording.
    5. Wait briefly while the etl files are merged.
    6. Retrieve mytrace.etl for analysis.

After the trace file has been collected, the trace file must be opened in WPA. This can be accomplished with a right-click on the .etl file and select *open with WPA*, or launch the WPA tool and use the *File* menu to open the .etl file.

Apply a WPA startup profile

A startup profile allows WPA to show a default view of the important information about the trace on each launch of WPA. The set of information displayed may be tailored for a specific scenario. It provides a view of the graphs that are keys to identify issues.

A CS startup profile is included in the same folder as this documentation. To apply the profile, follow these steps:

1. Go to the *Profiles* menu and select *Apply*
2. Select the CS.wpaProfile
3. Go to the *Profiles* menu and select *Save StartupProfile*

Once a startup profile is saved, the startup profile is persisted and will be used in subsequent launches of the WPA tool.

What is DRIPS?

In order to understand the details of CS analysis, you must first understand the meaning of DRIPS. DRIPS stands for Deepest Runtime Idle Platform State and is when the system is consuming the lowest amount of power possible, limited by the power floor. During CS, the system remains powered on to do certain tasks such as receiving emails, updating live tiles with fresh content, and receiving VoIP calls etc. When the system is performing those tasks, the system is not in DRIPS.

A system is considered to be in CS when the screen is off but the system remains powered on. A CS session is defined as the time when the screen is turned off to when the screen is turned back on. On CS entrance, the system goes through multiple phases to put the system into a low-power state. When the system is in the lowest-power state, we say the system is in DRIPS. When the system is not in DRIPS, it's in non-DRIPS. Therefore the total CS session time is summation of DRIPS and non-DRIPS time. Generally speaking, the higher the DRIPS% is, the longer the battery life is with a few exceptions. Our primary goal is to maximize DRIPS time on all platforms that support CS.

Understanding and manipulating the graphs

The graphs in the startup profile are keys to tell if the trace has issues and what caused those issues. This section covers how you can use the graphs to analyze the trace and what key information you can extract from each graph.

Each graph also has a table view which contains the data that was used to construct the graph. The view can be configured via the buttons located at the upper right-hand corner of the graph window. Figure 1 below shows a screenshot of the three views you can choose for each graph:

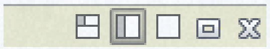


Figure 1. Graph and table views selection

Starting from the left, the first button is for displaying both graph and table, then graph only, then the last one is table only. The default view is graph only. This is important to note because the following sections require you to change the default view to obtain analysis information.

Platform Idle State graph

This graph shows the platform wide idle state plotted against time. On different platforms the numerical states may correspond to different states; platform specific documentation should provide a mapping. You should contact your hardware partner for which platform idle state is DRIPS for your SoC. Figure 2 below shows an example of the graph and a SoC that has State 3 as DRIPS and State 0 as active.

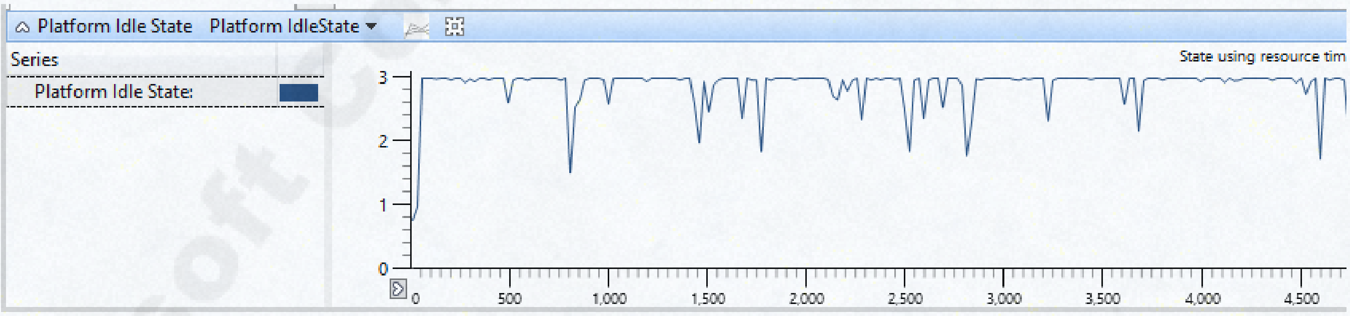
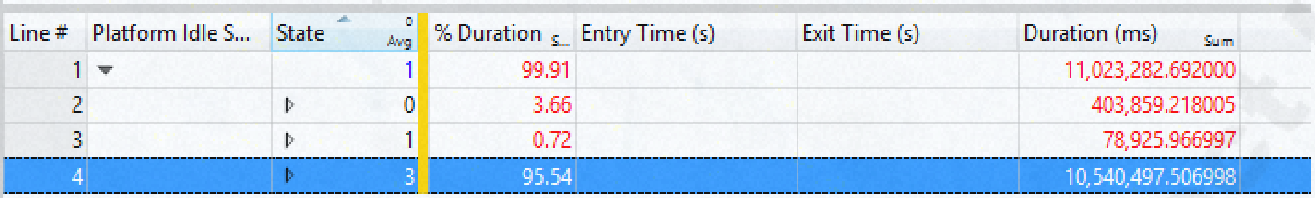


Figure 2. Platform Idle State graph view

You can use this graph to figure out the DRIPS% which is the percentage of time when the system is in DRIPS during the CS session. This is important data to know because it has a direct implication to battery life. If the DRIPS% is high, battery life is longer than if DRIPS% was lower in most cases. To obtain the DRIPS%, open the table view and drag the

% Duration column to filter on State. This will tell you the % of time the system was in each state. Figure 3 below shows the column configurations and a SoC that has 95.54 DRIPS%.



Line #	Platform Idle S...	State	% Duration	Entry Time (s)	Exit Time (s)	Duration (ms)
1	▾	1	99.91			11,023,282.692000
2		▷ 0	3.66			403,859.218005
3		▷ 1	0.72			78,925.966997
4		▷ 3	95.54			10,540,497.506998

Figure 3. Platform Idle State graph table view

Note that the platform state information shown on this graph includes the time from starting the trace to turning off the screen and the time after the screen is back on until the trace is stopped. It is recommended that you capture a trace that is long enough to make these numbers negligible.

DRIPS graph

This graph shows the components that are active during the trace period including activators, devices, and processes. You can use this graph to figure out the components that are active the longest which prevented the system from entering DRIPS. Figure 4 below shows a view of the graph and their respective active periods in colored regions.

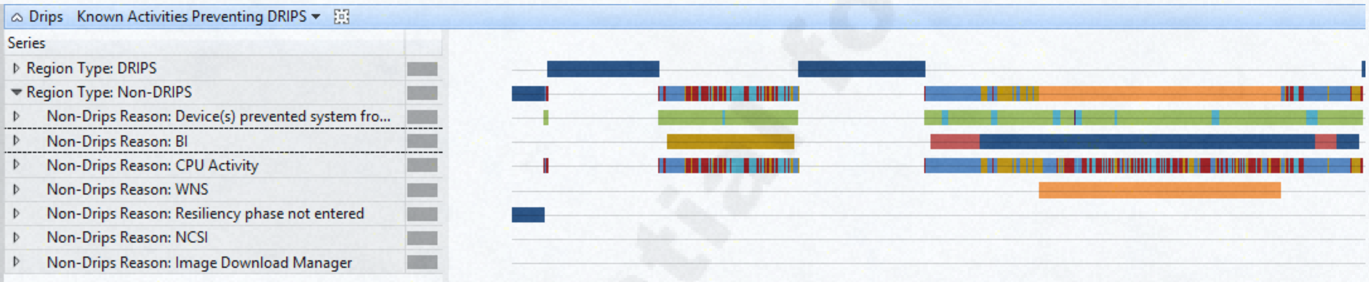
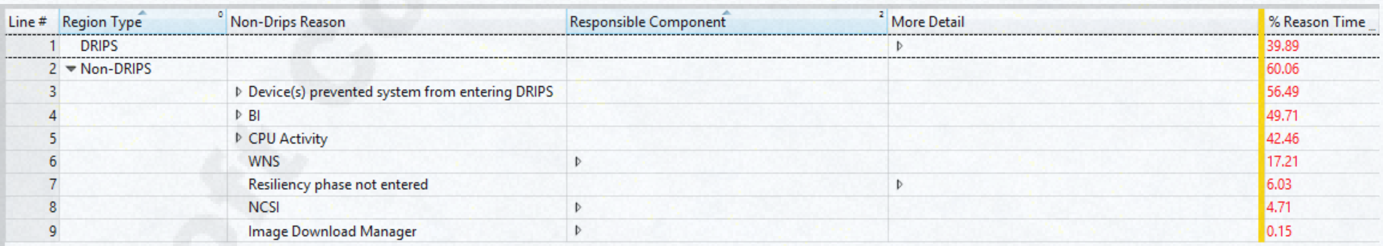

Drips Known Activities Preventing DRIPS
Series
Region Type: DRIPS
Region Type: Non-DRIPS
Non-Drips Reason: Device(s) prevented system fro...
Non-Drips Reason: BI
Non-Drips Reason: CPU Activity
Non-Drips Reason: WNS
Non-Drips Reason: Resiliency phase not entered
Non-Drips Reason: NCSI
Non-Drips Reason: Image Download Manager

Figure 4. DRIPS graph view

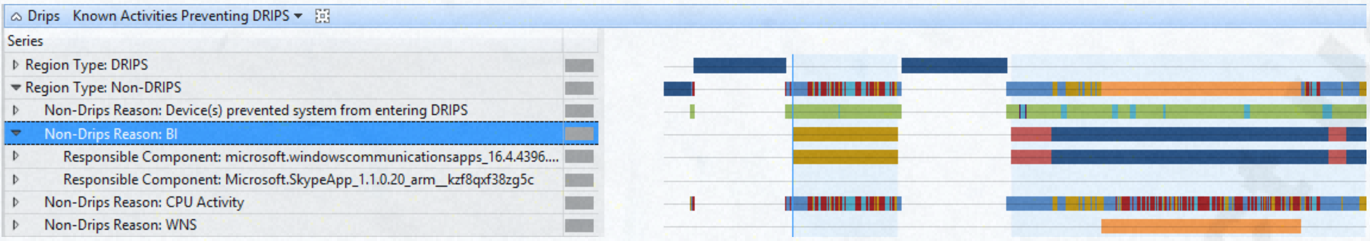
Activators are components that could take references and perform tasks while in CS. Some common ones are BI, WNS, and WU. You can use this graph to figure out the top active Activator during the CS session. This is important because a particular activator could be holding a reference for long periods of time which prevented the system from entering DRIPS. Components that are shown in this graph except for Devices and CPU Activity are activators. For example, Figure 4 above shows BI, WNS, NCSI, and Image Download Manager as activators. You can figure out the top activators by opening the table view and look at the *% Reason Time* column which shows the percentage of time the activator was active during the CS session. Figure 5 below shows an example where BI is the top activator with 49.71% active.



Line #	Region Type	Non-Drips Reason	Responsible Component	More Detail	% Reason Time
1	DRIPS			▷	39.89
2	▾ Non-DRIPS				60.06
3		▷ Device(s) prevented system from entering DRIPS			56.49
4		▷ BI			49.71
5		▷ CPU Activity			42.46
6		WNS	▷		17.21
7		Resiliency phase not entered		▷	6.03
8		NCSI	▷		4.71
9		Image Download Manager	▷		0.15

Figure 5. DRIPS graph table view

BI is a special activator because it provides broker services to apps to access system resources. When BI shows up as an active activator, you can expand the BI row and figure out which apps are causing BI to be active. You can use this graph to figure out the top active apps during the CS session. Figure 5 below shows an example of two apps (Windows Live and Skype) that are causing BI to be active and the Windows Live app was active a majority of the time.



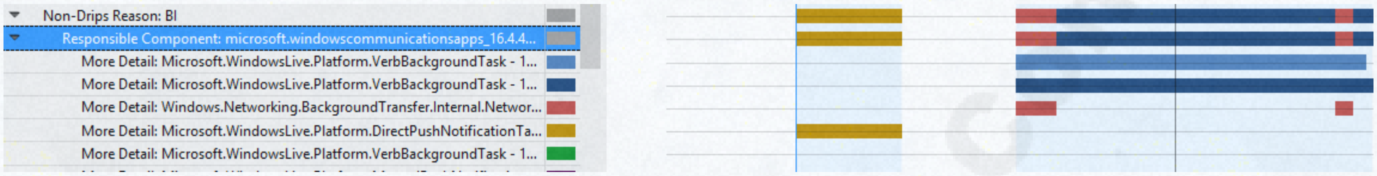
Drips Known Activities Preventing DRIPS

Series

- Region Type: DRIPS
- Region Type: Non-DRIPS
  - Non-Drips Reason: Device(s) prevented system from entering DRIPS
  - Non-Drips Reason: BI
    - Responsible Component: microsoft.windowscommunicationsapps_16.4.4396....
    - Responsible Component: Microsoft.SkypeApp_1.1.0.20_arm__kzf8qxf38zg5c
  - Non-Drips Reason: CPU Activity
  - Non-Drips Reason: WNS

Figure 6. Expanded BI view that shows active apps

To drill deeper into the apps, you can expand each app in the graph view and it will show you information on the individual tasks within the app that are keeping the app alive.

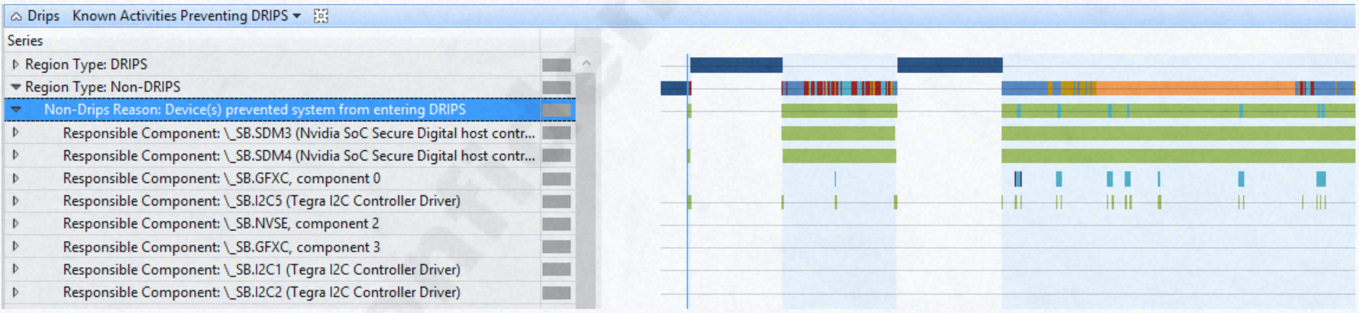


- Non-Drips Reason: BI
  - Responsible Component: microsoft.windowscommunicationsapps_16.4.4...
    - More Detail: Microsoft.WindowsLive.Platform.VerbBackgroundTask - 1...
    - More Detail: Microsoft.WindowsLive.Platform.VerbBackgroundTask - 1...
    - More Detail: Windows.Networking.BackgroundTransfer.Internal.Networ...
    - More Detail: Microsoft.WindowsLive.Platform.DirectPushNotificationTa...
    - More Detail: Microsoft.WindowsLive.Platform.VerbBackgroundTask - 1...

Figure 7. Expanded app view that shows active tasks

Besides activators, another possible reason that the system could not enter DRIPS is because some devices are actively running. The name of the devices is hardware-dependent. You can obtain the device names for your platform by looking at the ACPI DSDT table. Similar to system idle states, devices have states. Devices have different states ranging from D0 to D3. Device states are standardized across platforms.

You can use this graph to figure out the top active devices during the CS session. This is important because a particular device could be holding a reference for long periods of time which prevented the system from entering DRIPS. Keep in mind that some devices can be active because an Activator is running some tasks which could cause them to be active. The following graph shows an example of a trace where SDM3 and SDM4 are the top active devices.



Drips Known Activities Preventing DRIPS

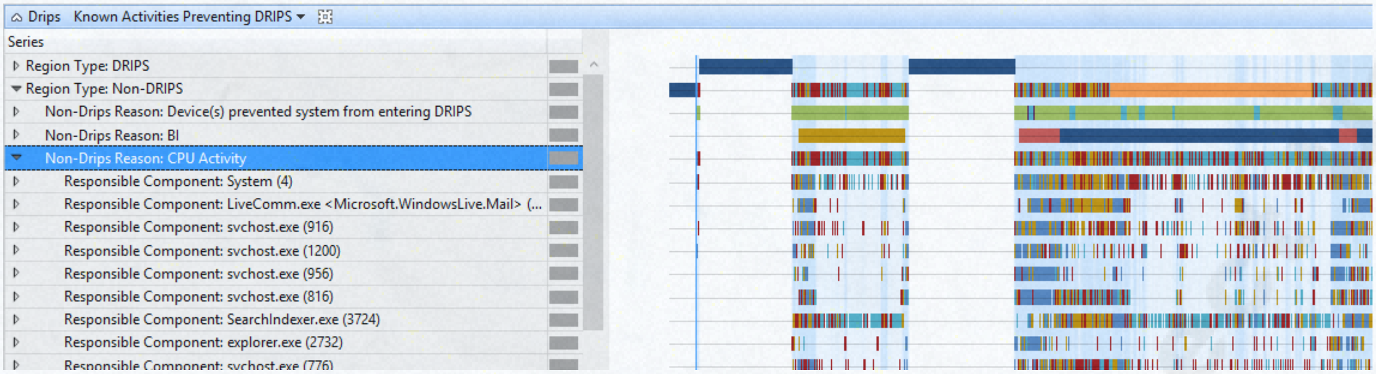
Series

- Region Type: DRIPS
- Region Type: Non-DRIPS
  - Non-Drips Reason: Device(s) prevented system from entering DRIPS
    - Responsible Component: _SB.SDM3 (Nvidia SoC Secure Digital host contr...
    - Responsible Component: _SB.SDM4 (Nvidia SoC Secure Digital host contr...
    - Responsible Component: _SB.GFXC, component 0
    - Responsible Component: _SB.I2C5 (Tegra I2C Controller Driver)
    - Responsible Component: _SB.NVSE, component 2
    - Responsible Component: _SB.GFXC, component 3
    - Responsible Component: _SB.I2C1 (Tegra I2C Controller Driver)
    - Responsible Component: _SB.I2C2 (Tegra I2C Controller Driver)

Figure 8. Expanded Devices view that shows active devices

You can figure out the top devices by opening the table view and look at the *% Reason Time* column which shows the percentage of time each device was active during the CS session.

Besides devices and activators, another possible reason that the system could not enter DRIPS is because there are some processes running that are causing the CPU to be active. CPU Activity is less common of a problem compared to activators and devices. You can see the active processes by expanding the CPU Activity row. Figure 9 below shows an example of the view.



Drips Known Activities Preventing DRIPS

Series

- Region Type: DRIPS
- Region Type: Non-DRIPS
  - Non-Drips Reason: Device(s) prevented system from entering DRIPS
  - Non-Drips Reason: BI
  - Non-Drips Reason: CPU Activity
    - Responsible Component: System (4)
    - Responsible Component: LiveComm.exe <Microsoft.WindowsLive.Mail> (...
    - Responsible Component: svchost.exe (916)
    - Responsible Component: svchost.exe (1200)
    - Responsible Component: svchost.exe (956)
    - Responsible Component: svchost.exe (816)
    - Responsible Component: SearchIndexer.exe (3724)
    - Responsible Component: explorer.exe (2732)
    - Responsible Component: svchost.exe (776)

Figure 9. Expanded CPU Activity view that shows active processes

Determining if the trace is worth investigation

We use DRIPS% as the key metrics to determine if a given trace exhibits good or bad battery life during a CS session. The DRIPS% is the percentage of time during the CS session when the system is in DRIPS. Generally speaking, high DRIPS% translates to longer battery life. You can use the Platform Idle State graph to obtain the DRIPS% of the trace. Here is some general guidance on how to evaluate the DRIPS%.

Table 1. DRIPS% evaluation guide

DRIPS %	Evaluation
98 – 100	Excellent
95 – 97.9	Very good
90 – 94.9	Good
80 – 89.9	Fair
< 80	Poor

There are some exceptional cases where even DRIPS% is high, the battery life is poor. For example, a DRIPS% of 95 or higher can result in less than 3 days of battery life. Those cases are likely due to hardware problems and should be investigated further to understand the root cause.

Identifying the Key Connected Standby Problems

There are several problems which may prevent a system from consistently entering DRIPS. A system cannot enter DRIPS if there are any tasks running that require the system to stay active. Similarly, the system cannot enter DRIPS if there are any SoC devices or connected devices powered on. The analysis steps below will walk you through the process to figure out which tasks or devices are causing non-DRIPS time.

1. The first thing to look at is resiliency activations. Look at the DRIPS graph and figure out the top active Activators. If there are none, skip to step 4.
2. Pick the top active activator and get the % of active time by looking at the graph table.
3. If the % of active time is significant, you should notify your Microsoft contact to understand why the activator is preventing the system from going into DRIPS. See the Reporting an Issue section for information on how to contact Microsoft. If the top Activator is BI, you should figure out the active app that is causing BI to be active. You can then supply this information to Microsoft directly and figure out why those apps are preventing the system from going into DRIPS.
4. If the % of active time by Activators isn't significant or not present, the next thing to look for is devices. Look at the DRIPS graph and figure out the top active devices and the % of time they're active.

5. If the % of active time is significant, it’s indicative that device is the reason why the system is not going into DRIPS. If you’re a hardware manufacturer, you should contact your device vendors to understand why they are preventing the system from going into DRIPS. For others, you should submit a bug report to Microsoft for further diagnosis of the issue. See the Reporting an Issue section for information on how to contact Microsoft.
6. If neither Activators nor devices present a significant problem, another possible culprit are the processes that are exempted by the DAM that can run during CS. These processes could be consuming resources which prevent the system from going into DRIPS. To see the list of processes, you can use the DRIPS graph to tell which processes were running for a long time. If the process was active for a long period of time, you should submit a bug report to Microsoft. See the Reporting an Issue section for information on how to contact Microsoft.

More detail diagnostics using Generic Events

Some components provide instrumentation that allows you to further drill down into the issues using WPA. The *Generic Events* graph shows the list of events that were triggered by devices, apps, and other components in the system.

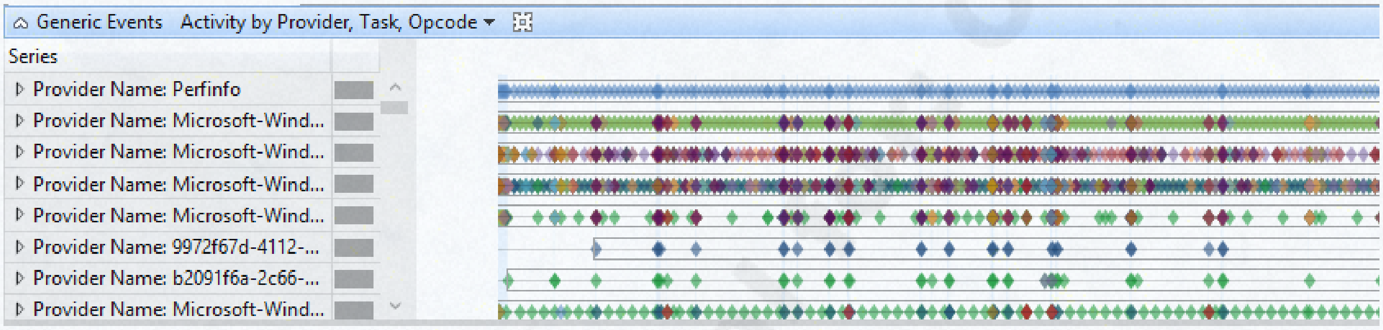
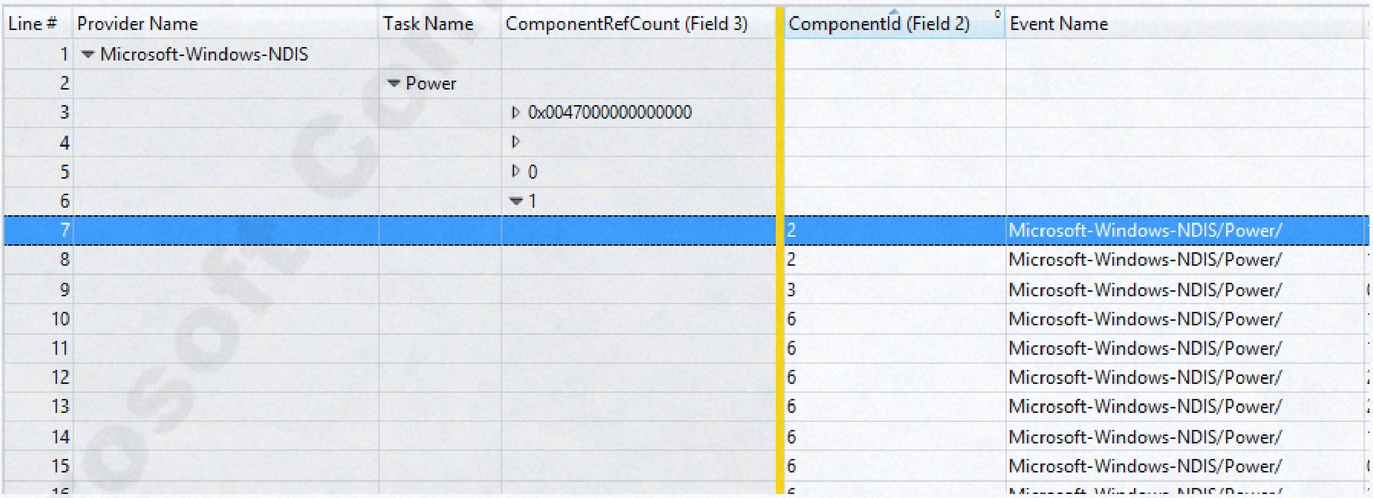

Generic Events
Activity by Provider, Task, Opcode
Series
Provider Name: Perfinfo
Provider Name: Microsoft-Wind...
Provider Name: Microsoft-Wind...
Provider Name: Microsoft-Wind...
Provider Name: Microsoft-Wind...
Provider Name: 9972f67d-4112-...
Provider Name: b2091f6a-2c66-...
Provider Name: Microsoft-Wind...

Figure 10. Generic Events graph view

You can use this graph to drill down into the specific component that caused the issue. For example, if you identified that the key culprit is the networking stack, you can drill down into which component in the networking stack could be holding resource references which is preventing the system from going to DRIPS. To do that, you can use the *Generic Events* table and filter to focus on the NDIS (Network Driver Interface Spec) component. In the table view under the *Power* category, make a note of the top components with the highest number of ComponentRefCount = 1 counts. Note that the column headers do not show until you expand it to the appropriate level in the tree. The following table shows an example of the table column configurations.



Line #	Provider Name	Task Name	ComponentRefCount (Field 3)	ComponentId (Field 2)	Event Name
1	▼ Microsoft-Windows-NDIS				
2		▼ Power			
3			▷ 0x0047000000000000		
4			▷		
5			▷ 0		
6			▼ 1		
7				2	Microsoft-Windows-NDIS/Power/
8				2	Microsoft-Windows-NDIS/Power/
9				3	Microsoft-Windows-NDIS/Power/
10				6	Microsoft-Windows-NDIS/Power/
11				6	Microsoft-Windows-NDIS/Power/
12				6	Microsoft-Windows-NDIS/Power/
13				6	Microsoft-Windows-NDIS/Power/
14				6	Microsoft-Windows-NDIS/Power/
15				6	Microsoft-Windows-NDIS/Power/

Figure 11. Generic Events graph table view

The screenshot above shows a list of components in the networking stack that took references which prevented the system from going into DRIPS. Each row represents a ComponentId taking a reference. The Component Id is the number assigned to the component based on an enumeration. To figure out what Component x is, you need to work with your Microsoft contact and figure out what component x maps to in order to identify the root cause and potential fix.

Reporting an Issue

To report a CS issue, you should work with your Microsoft contact (if you have one) or send an email to the Microsoft team (csfeedback@microsoft.com) directly with the appropriate information to carry forward with the investigation.

You should include the following information when you send the bug report:

Field	Value
Manufacturer/Model	<The model and make of the system experiencing issues>
Trace File	<The ETL trace>
CS Duration (seconds)	<This is the total CS session time in seconds>
DRIPS %	
Top Activator (and % active)	<Top active activator. If this is BI, also include the app name. You can obtain the % active by taking the active duration divide by the total CS duration>
Top Device (and % active)	<Top active device. You can obtain the % active by taking the active duration divide by the total CS duration>
Other Additional Information	