# Read Me

Feb 8, 2013

## Key Files

- *Install.cmd* - Batch file to install WPT.
- *trace_start.cmd* – Batch file to start trace collection using Xperf. It enables all providers necessary for power management analysis using WPA.
- *trace_end.cmd* – Batch file to stop trace collection using Xperf.
- *PowerManagement_Profile.wprp* – Profile file for trace collection using WPRUI.
- *PPM.wpaProfile* – WPA profile for Processor Power Management analysis.
- *CS.wpaProfile* – WPA profile for Connected Standby analysis.
- *Trace Analysis for Connected Standby Issues.docx* – Document that describes the techniques of analyzing a Connected Standby trace.
- *ReadMe.docx* – This file.

## Installation

Follow the steps outlined below to install the Windows Performance Toolkit (WPT):

1. If there is an older version of the toolkit already installed on the machine, uninstall it by going to Control Panel-> Programs and Features.
2. Open an elevated cmd prompt.
3. Change to the directory containing the files listed above.
4. Run *install.cmd*.
5. Copy *trace_end.cmd*, *trace_start.cmd* and the profile files to the WPT install location. The default install location for WPT is C:\Program Files\Windows Kits\8.1\Windows Performance Toolkit.

## Collecting a Trace

Power Management Analysis of a Windows PC involves collecting a software (ETW) trace. There are several methods for collecting a trace for analysis with these tools.

### Using the provided Xperf batch files

1. Copy *trace_start.cmd* and *trace_end.cmd* to the system in question.
2. Run *trace_start.cmd* from an elevated command prompt in the directory containing xperf.
3. Perform the desired scenario steps.
4. 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.

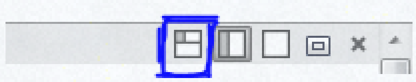
## Using Windows Performance Recorder UI

1. Copy *PowerManagement_Profile.wprp* to the system in question.
2. Run wprui.exe from the WPT install location with administrative privileges to start Windows Performance Recorder and check if “Power Management” appears under “Custom Measurements” in the list of profiles.
    a. If it does, proceed to step 3.
    b. If not, click “Add Profiles…” and open PowerManagement_Profile.wprp which you copied in step 1.
3. Press Start to begin recording the trace.
4. While recording, perform whatever workload is desired.
5. When the workload is finished, or you wish to cease recording, press Save.
6. Choose a file name on the dialog that appears.
7. Enter at least one letter into the description field and press Save.
8. Wait for the progress bar to complete and then press OK.
9. If Windows Performance Recorder is till Recording press Cancel.
10. Retrieve your saved trace file for analysis.

## Viewing a Trace

For viewing and analyzing Connected Standby traces, see the *Trace Analysis for Connected Standby Issues* document.

1. Open Windows Performance Analyzer (WPA) by running wpa.exe from the appropriate directory for the analyzing computer’s architecture which can differ from the computer on which the trace was collected. Note that wpa.exe is available for x86 and x64 only.
2. In the WPA menu, Click **File->Open** and select a trace file.
3. Click **Profiles->Apply** and select either *PoFx.wpaProfile* or *CS.wpaProfile* (which is present in the same folder this document is in) to open it depending on which set of graphs you wish to view. See the types of graphs available in each of these profiles below.
4. By default, the analysis view only shows the graphs. You can click the following button to show the summary table that is used to plot the graph.

    Note that the height may be too small to show the summary table by default, you might need to drag and reduce the graph height. You may also re-arrange items in the summary table which might affect the graph. To display the original graph again, you can return the table to the original configuration or re-apply the profile.

5. You can add other graphs to the current analysis view from Graph Explorer by following these steps:
    a. Expand a graph category in Graph Explorer.
    b. Select the graph you want to add and drag it to the analysis view pane.

## Types of Graphs in the CS Profile

See the *Trace Analysis for Connected Standby Issues* document for details.

## Types of Graphs in the PPM Profile

**CPU Idle States**: This graph is the same as that found in the PoFX profile.

**Processor Utilization**: A plot of % processor busy time as determined by the Windows PPM engine.

**Processor Performance**: A plot of the % of nominal performance delivered by the processor.

**Processor Utility**: A value between 1-10000 and is the product of Processor Performance and Processor Utilization.

**Processor Frequency**: This graph represents the processor's actual frequency. This graph can be the same as the processor performance unless the platform implements a performance metric that is different from frequency.

**Processor Constraints**: This graph shows constraints on the processor frequency. The sources of constraints shown are detailed below.

*Thermal*: Performance cap % from a thermal zone.

*BIOS*: Performance cap % through the performance control interface (P-state, CPPC, PCCv1, or PEP)

**Processor Parking State**: Current parked/unparked state of the processor.

**Core Parking Instantaneous Concurrency:** This visualization is a plot of the concurrency of the workload (i.e the parallelism of the workload measured as number of cores that the workload is employing simultaneously) over time.

**Core Parking Concurrency**: The processor power management engine makes core parking decisions based on the computed concurrency (among other things). It determines the computed concurrency by studying the instantaneous concurrency over the performance check duration.

**Core Parking Cap State**: BIOS caps on core parking through the logical processor idling interface.

## Other Graphs

In addition to the above graphs, the following other graphs can be selected from Graph Explorer.

**ThermalZone Device throttle**: This graph plots performance throttling applied to each thermal zone in the system. If the throttle is 100%, it implies there is not performance throttling. If it is 75% it implies the thermal zone has been capped to 75% performance.

**ThermalZone Temperature**: This graph plots the temperature reported by each thermal zone in the system.

**System Latency Tolerance**: This graph plots the current system latency tolerance specified by the operating system.

**Rail Voltage MilliVolt *(not supported on all platforms)***: This graph plots the voltages of any voltage rails exposed by the Power Engine Plugin

**Component Frequency KHz *(not supported on all platforms)*:** This graph plots the frequencies of any frequency rails exposed by the Power Engine Plugin

**Devices Preventing Platform Idle *(not supported on all platforms)*:** This visualizations shows information reported by the Power Engine plugin on which device D-states and component F-states are requirements for entering a platform idle state by plotting at each instant of time, the devices that are preventing a specific platform idle state.

**CPU Idle States**: This graph shows the idle state of each CPU plotted against time. When CPUs are in state 0, they are active and doing work. When in a state greater than 0, they are idle. A deeper state usually indicates greater power savings.

On different platforms the state indices may correspond to different idle states.

**Platform Idle State**: 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. *Note for this graph to be enabled, the PEP must implement platform idle states and report them to the OS.*

**CPU Usage (Precise)**: This graph shows the CPU usage by process and thread over time. This allows for the identification of specific sources of software activity.

**Device D-state**: This graph shows for each device on the system, the current Dx state. Devices are in the D0 state when in use. Both devices registered with the power frame work (Type: PoFx) and ones not registered with the framework (Type:Non-PoFx) are displayed. PoFx devices can leave D0 to a Dx (x>0) state when the framework provides the “Power Not Required” message. Non PoFx devices added after the start of the trace may not display complete state information.

**PoFx Component F-state**: This graph shows for each device, the components it has and plots the F-state of each device component as a function of time.

**PoFx Device Power Requirement:** For each device registered with the power framework, this graph plots the Power Required/Power Not Required status as determined by the power framework. When a device's power is required, the graph shows state 1. When a device's power is not required, the graph shows state 0.

Note that a device is only allowed to leave the D0 state when both the framework declares device power is not required.

**PDC Notification Phase**: This visualization shows the duration of each phase of a connected standby transition as coordinated by the Power Dependency Coordinator (PDC).

The phases involved in connected standby entry are –

- Connected Sessions Check
- Presence
- PLM
- DAM
- Low Power Epoch
- Resiliency Notification
- Resiliency

The duration of each phase is marked by a block timeline (that may look like markings on a line depending on the zoom level). All of the phases are typically very fast (order of ms). The Low Power Epoch phase take ~5s due to a built in 5s delay.

Under each phase, the visualization also shows the source of activity in each phase.

**LP Events**: This visualization shows the periods of time the system is Active, in Connected Standby, Full Screen Video Playback or Low Power Audio scenarios. It also shows power button events and the display off event.

**PDC Resiliency Activity**: This visualization shows the "Activators" that disable resiliency technologies. There are 4 resiliency technologies

- Idle Resiliency
- IO Resiliency
- DAM Resiliency
- Network Quiet Mode (NQM) Resiliency

Each one of these resiliency technologies can be disabled by components registered as "Activators". When the system is not in connected standby the PDC itself acts as an activator to disable the resiliency technologies. When the system is in connected standby we should typically none or very little intermittent "Activator" activity.

In the visualization, whenever an Activator sets the value 1, the resiliency technology is disabled