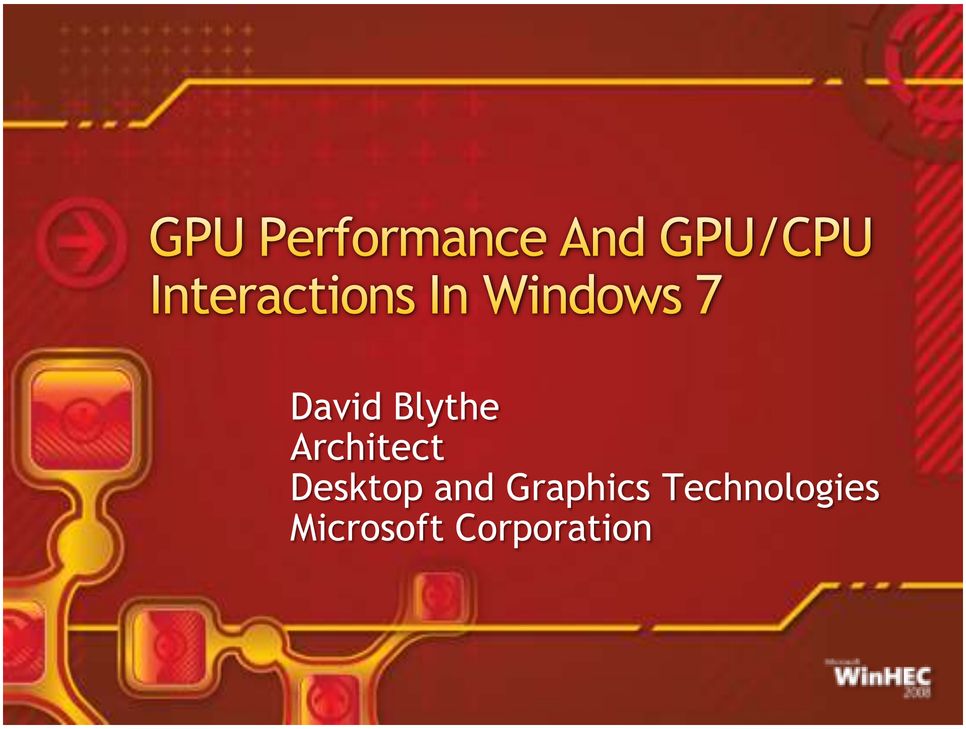
Microsoft WinHEC 2008

Advancing the Platform

Microsoft

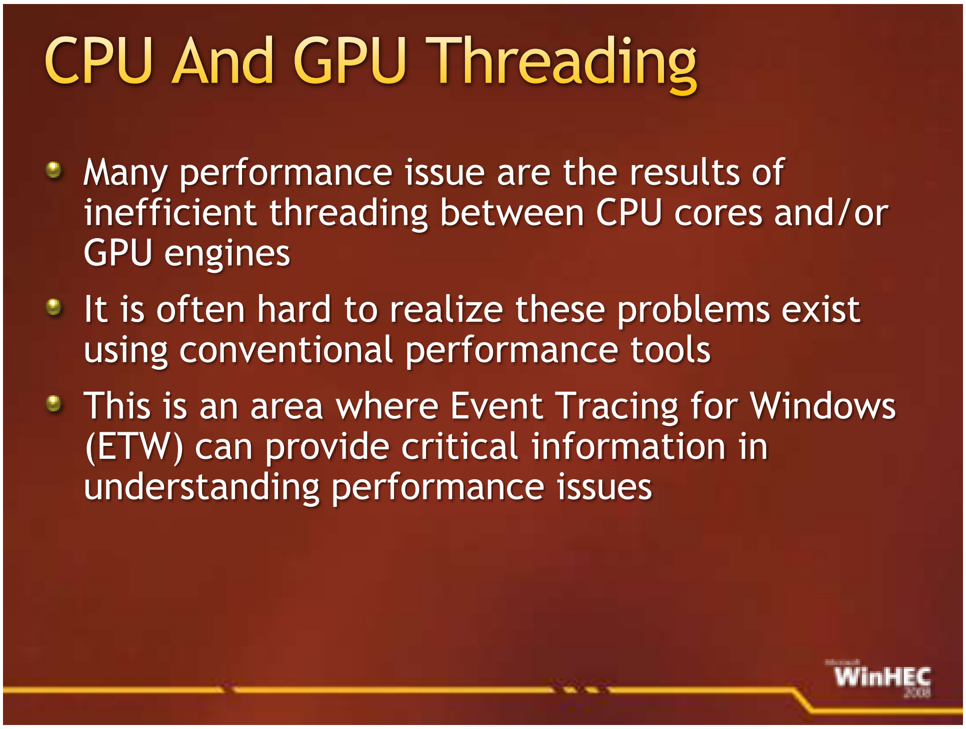


GPU Performance And GPU/CPU Interactions In Windows 7

David Blythe
Architect
Desktop and Graphics Technologies
Microsoft Corporation

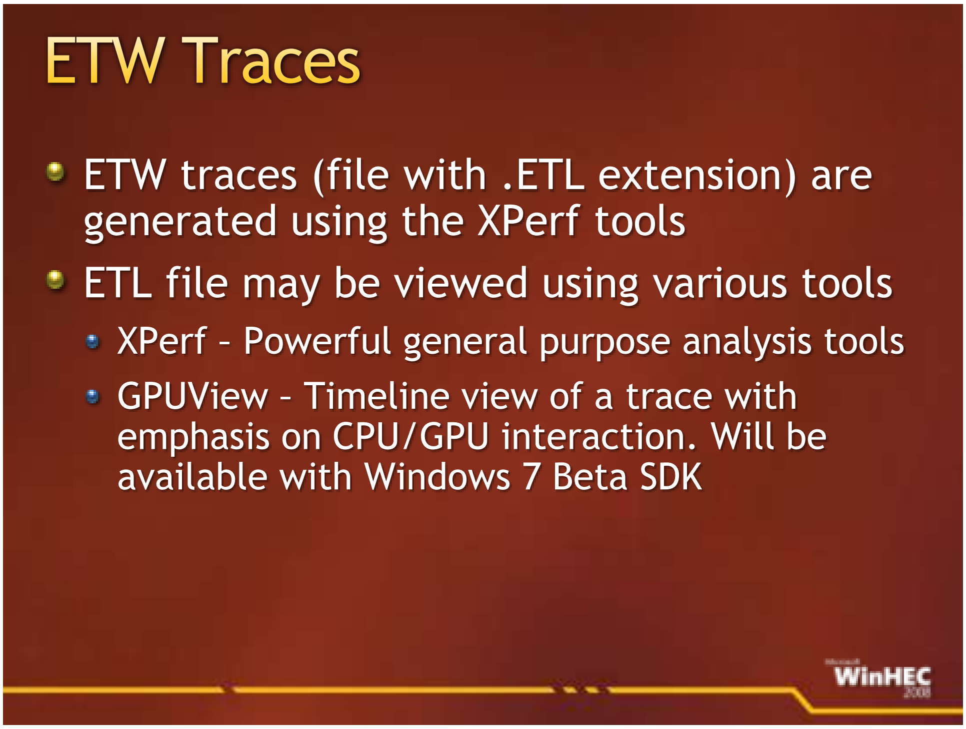
Microsoft WinHEC 2008

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.



CPU And GPU Threading

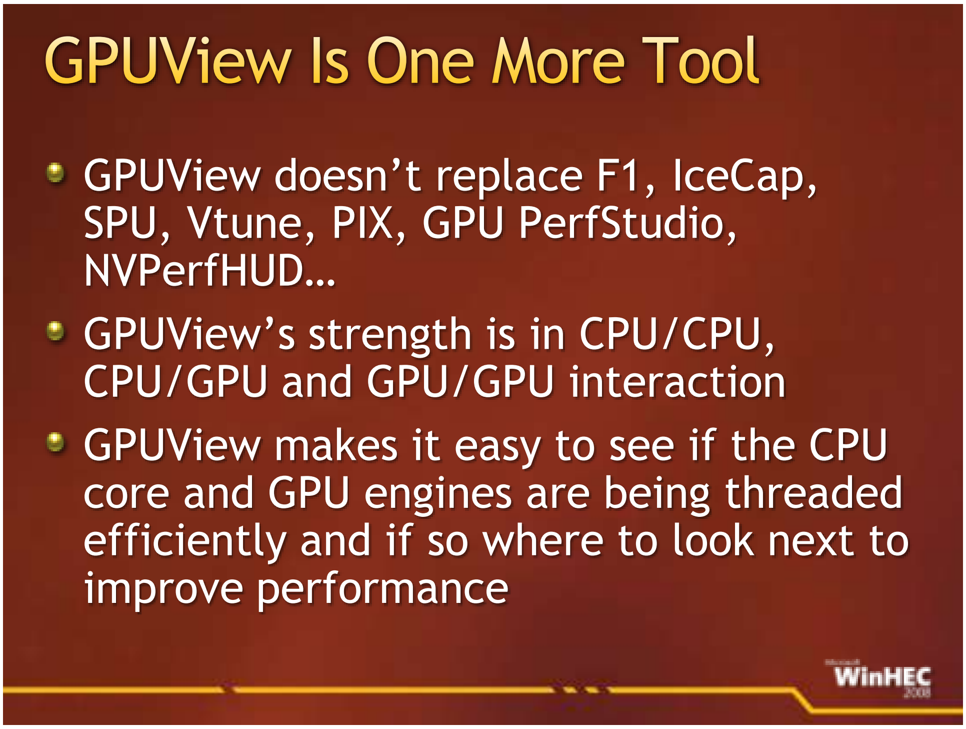
- Many performance issue are the results of inefficient threading between CPU cores and/or GPU engines
- It is often hard to realize these problems exist using conventional performance tools
- This is an area where Event Tracing for Windows (ETW) can provide critical information in understanding performance issues



ETW Traces

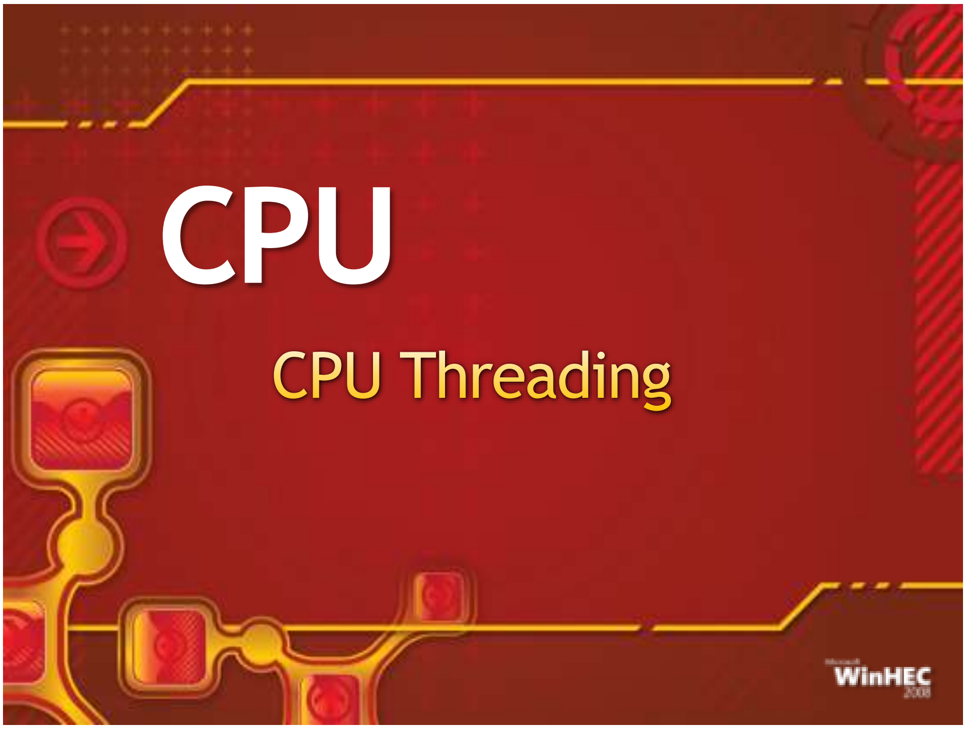
- ETW traces (file with .ETL extension) are generated using the XPerf tools
- ETL file may be viewed using various tools
  - XPerf - Powerful general purpose analysis tools
  - GPUView - Timeline view of a trace with emphasis on CPU/GPU interaction. Will be available with Windows 7 Beta SDK

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.



GPUView Is One More Tool

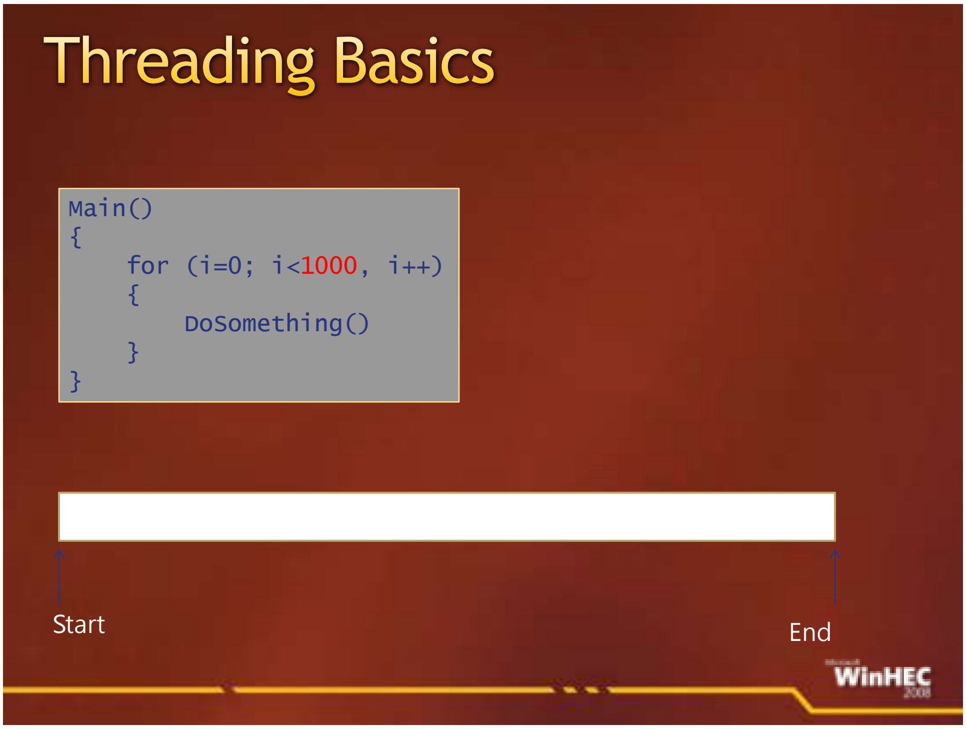
- GPUView doesn’t replace F1, IceCap, SPU, Vtune, PIX, GPU PerfStudio, NVPerfHUD…
- GPUView’s strength is in CPU/CPU, CPU/GPU and GPU/GPU interaction
- GPUView makes it easy to see if the CPU core and GPU engines are being threaded efficiently and if so where to look next to improve performance



CPU

CPU Threading

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

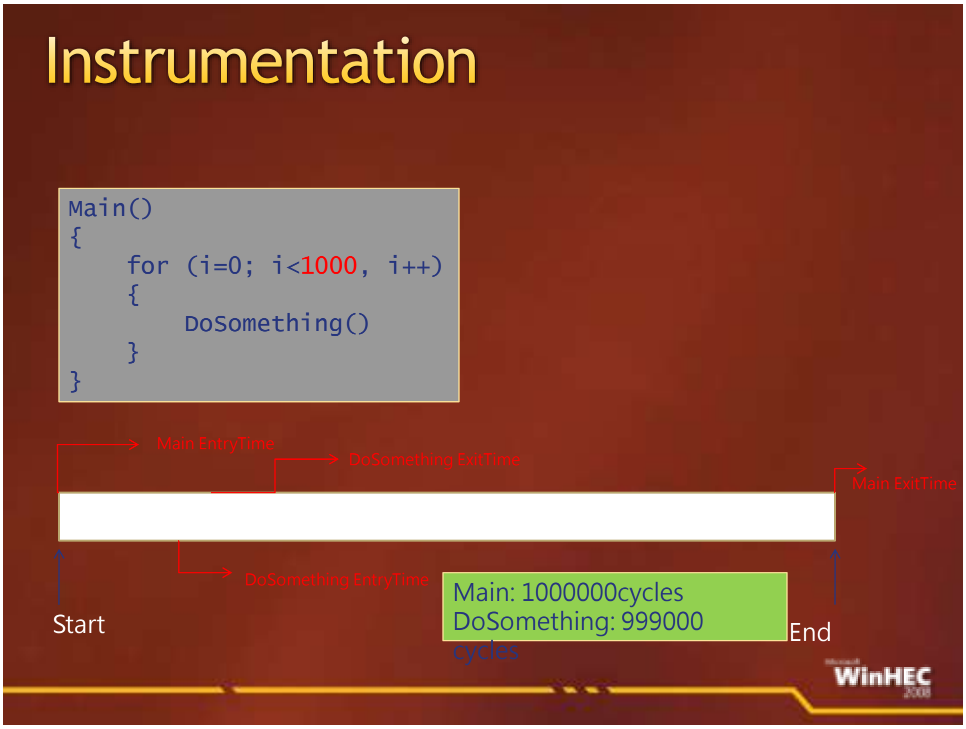


Threading Basics

```
Main()
{
    for (i=0; i<1000, i++)
    {
        DoSomething()
    }
}
```

Start

End



Instrumentation

```
Main()
{
    for (i=0; i<1000, i++)
    {
        DoSomething()
    }
}
```

Main EntryTime

DoSomething ExitTime

Main ExitTime

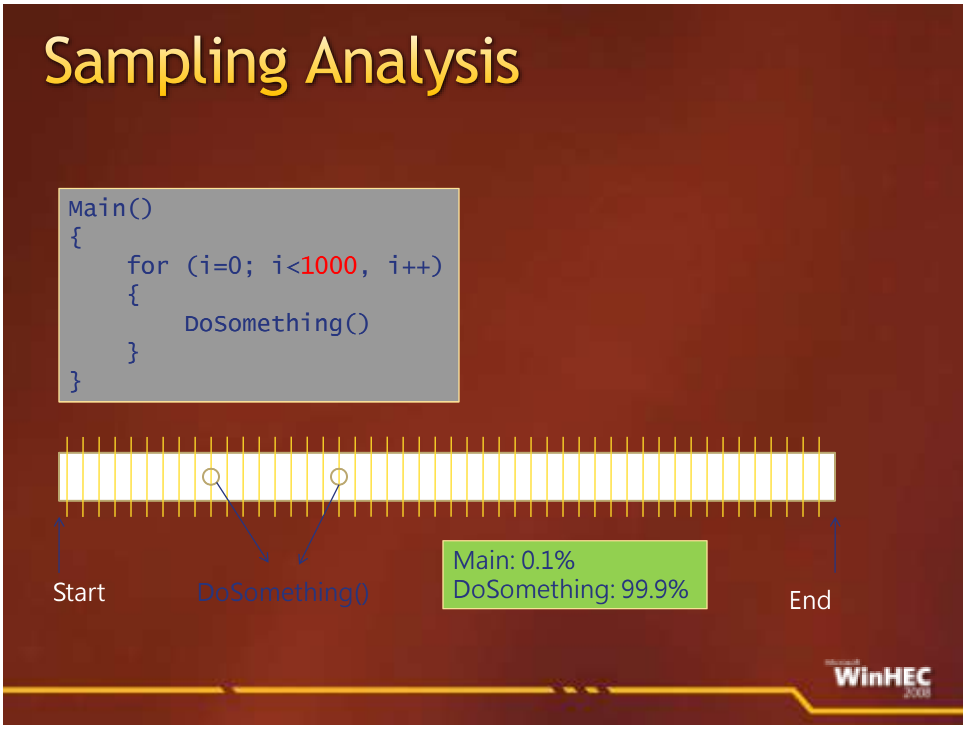
DoSomething EntryTime

Start

Main: 1000000cycles
DoSomething: 999000 cycles

End

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.



Sampling Analysis

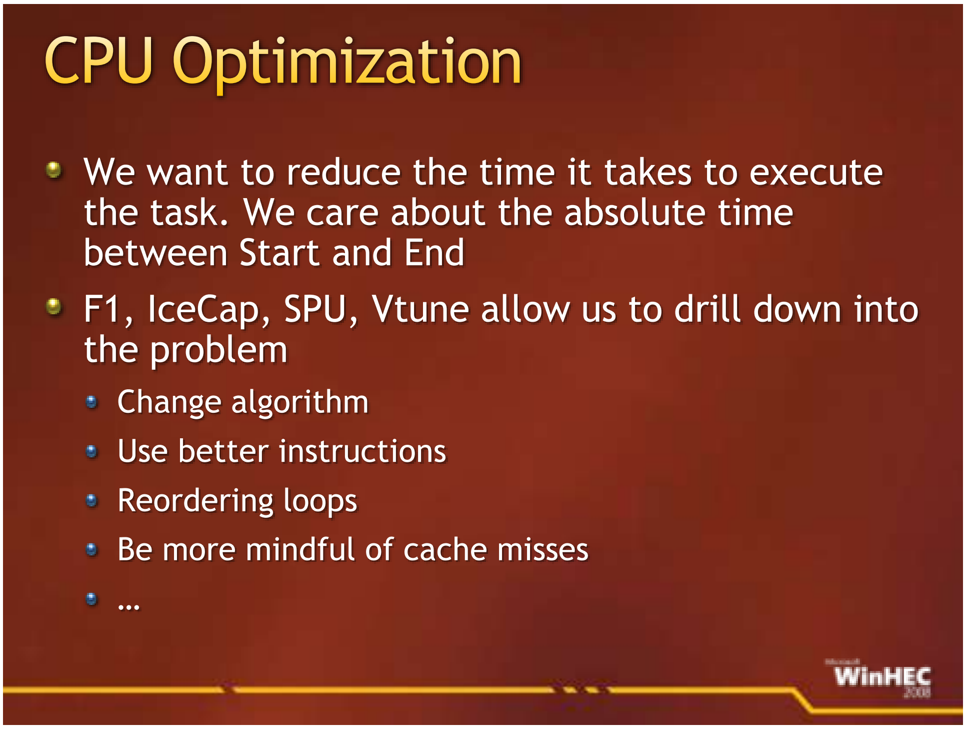
```
Main()
{
    for (i=0; i<1000, i++)
    {
        DoSomething()
    }
}
```

Start

DoSomething()

Main: 0.1%
DoSomething: 99.9%

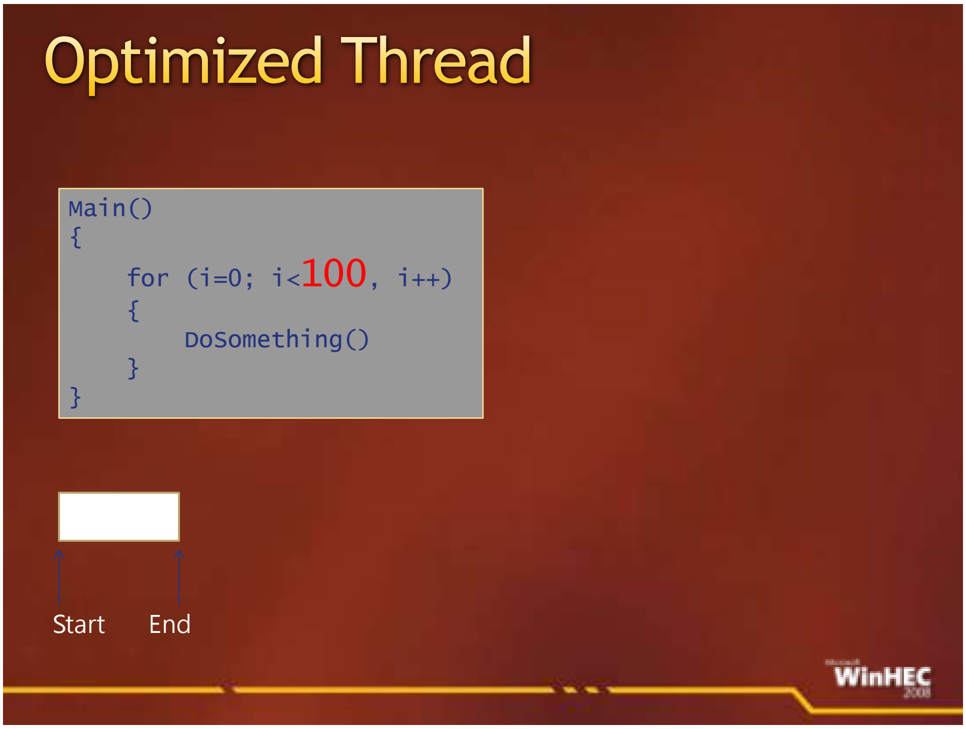
End



CPU Optimization

- We want to reduce the time it takes to execute the task. We care about the absolute time between Start and End
- F1, IceCap, SPU, Vtune allow us to drill down into the problem
  - Change algorithm
  - Use better instructions
  - Reordering loops
  - Be more mindful of cache misses
  - …

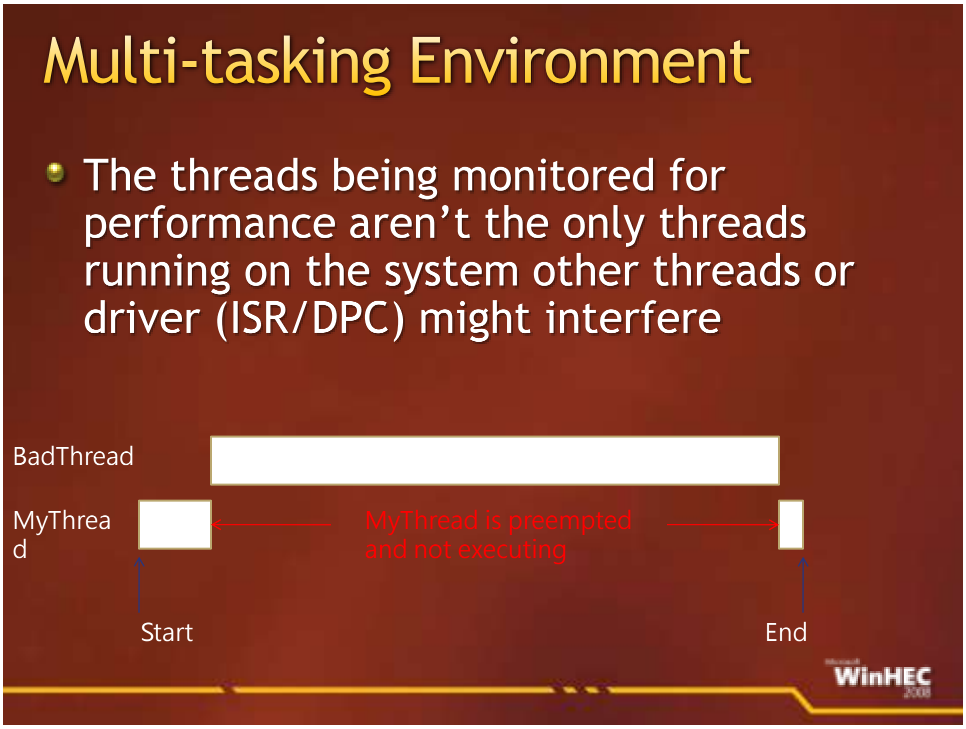
© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.



Optimized Thread

```
Main()
{
    for (i=0; i<100, i++)
    {
        DoSomething()
    }
}
```

Start
End
WinHEC 2008



Multi-tasking Environment

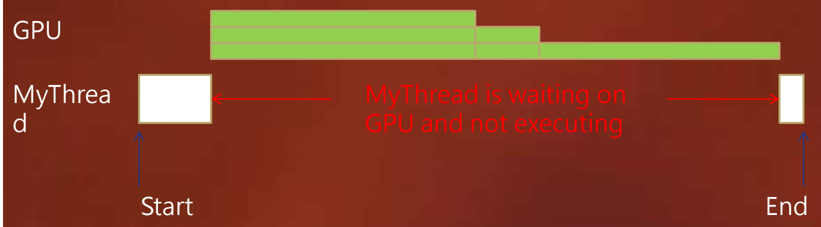
- The threads being monitored for performance aren’t the only threads running on the system other threads or driver (ISR/DPC) might interfere

BadThread
MyThread
MyThread is preempted and not executing
Start
End
WinHEC 2008

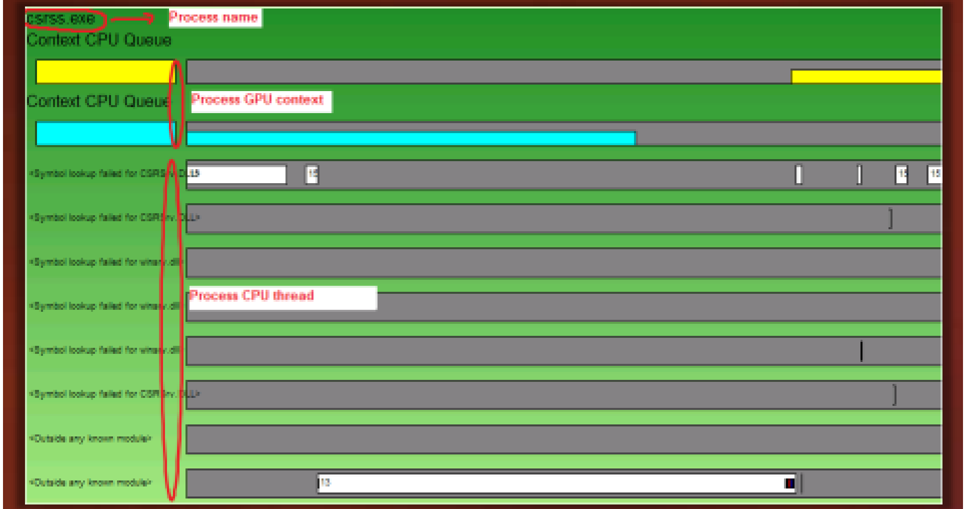
© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Device Interaction

- A thread might interact with other threads or with a device and may be put into a wait state waiting for something to happen on another thread or device



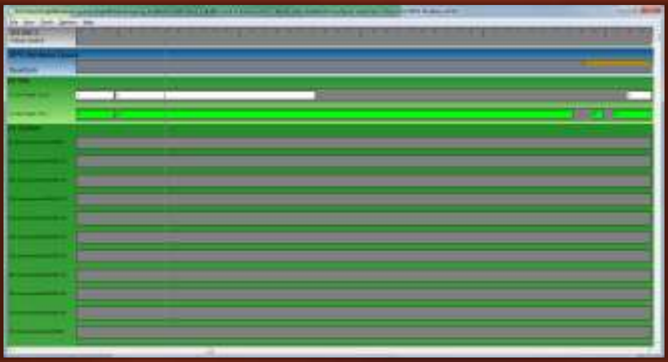
GPUView Process View


csrss.exe
Process name
Context CPU Queue
Context CPU Queue
Process GPU context
<Symbol lookup failed for CSRSrv.DLL>
<Symbol lookup failed for CSRSrv.DLL>
<Symbol lookup failed for winsrv.dll>
Process CPU thread
<Symbol lookup failed for winsrv.dll>
<Symbol lookup failed for winsrv.dll>
<Symbol lookup failed for CSRSrv.DLL>
<Outside any known module>
<Outside any known module>

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Idle Thread

- Every CPU core executes a particular thread at any given time; When Idle, a core executes the idle thread



WinHEC 2008

CPU Core

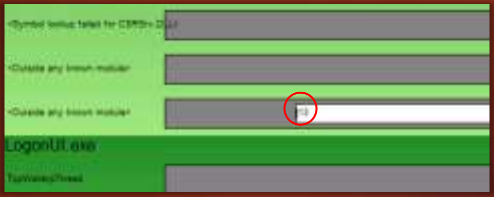
- Every CPU core gets assigned a color, threads are colored from the core they are executing on

WinHEC 2008

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Thread Priority

- The priority at which a thread segment is executing is displayed in the left portion of that segment; If the segment isn't large enough, zoom in

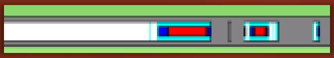


<Symbol lookup failed for CSRSS> [illegible]
<Outside any known module>
<Outside any known module>
LogonUI.exe
TpWaiterThread

WinHEC 2008

Thread Execution

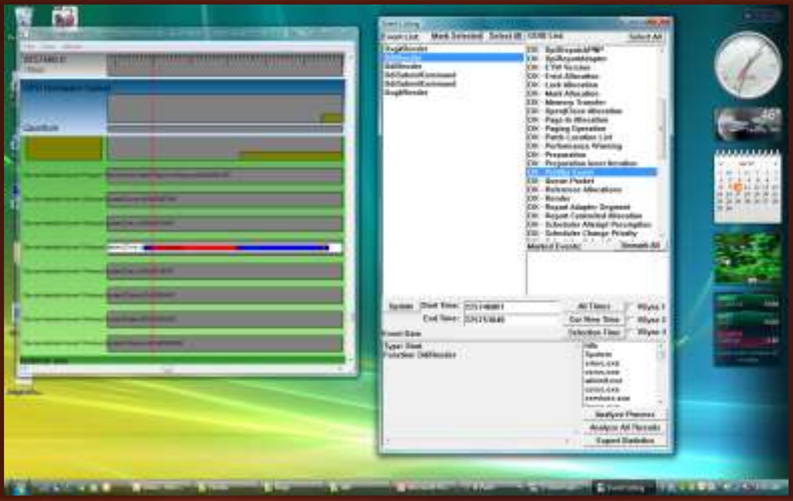
- Thread segments are colored to help identified what the thread is currently doing
  - Light blue: Thread in kernel mode
    (only if syscall events were enabled as tracing option, not available on MP system)
  - Dark blue: Thread in dxgkrnl
  - Red: Thread in KMD



WinHEC 2008

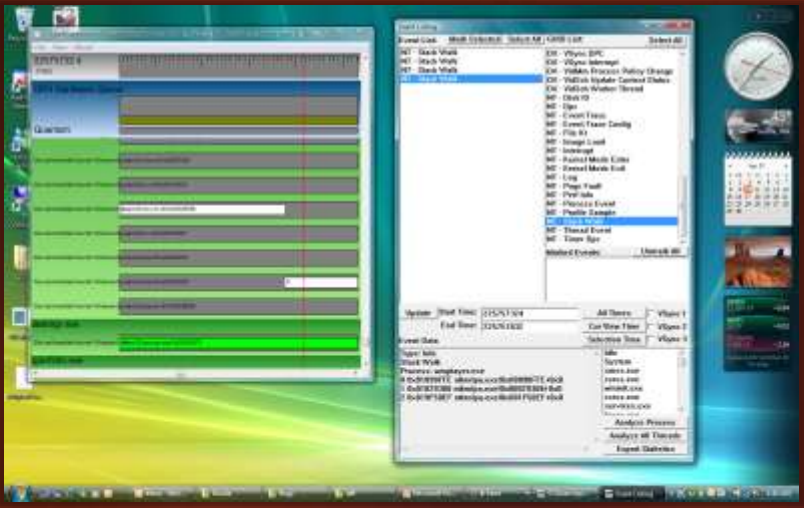
© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Thread Profiler Events



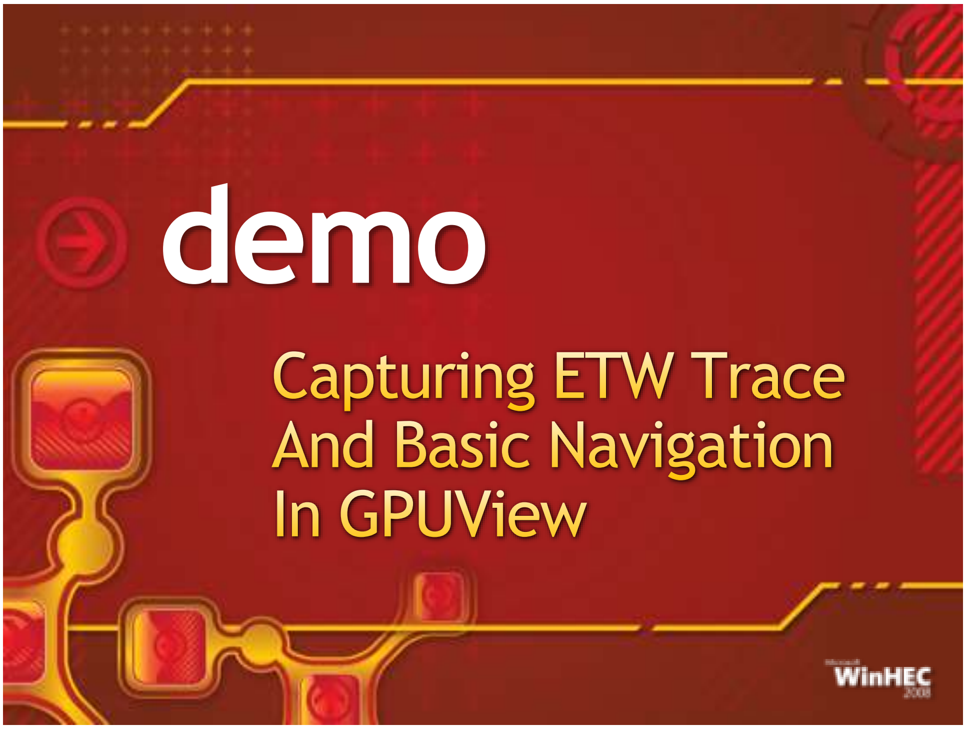
WinHEC 2008

Context Switch Stack



WinHEC 2008

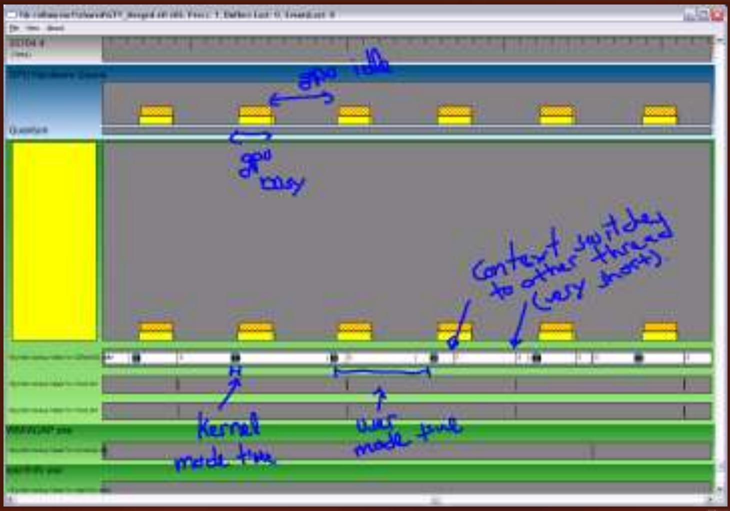
© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.



demo

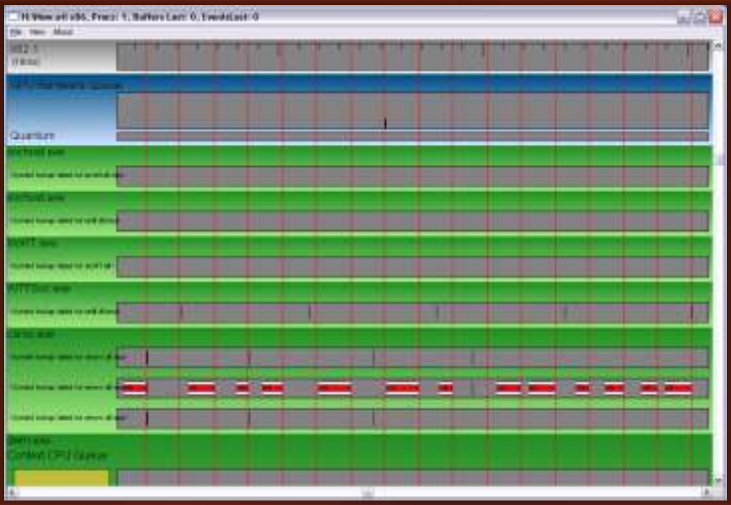
Capturing ETW Trace And Basic Navigation In GPUView

Example: CPU Bound Game



© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Example: Interference

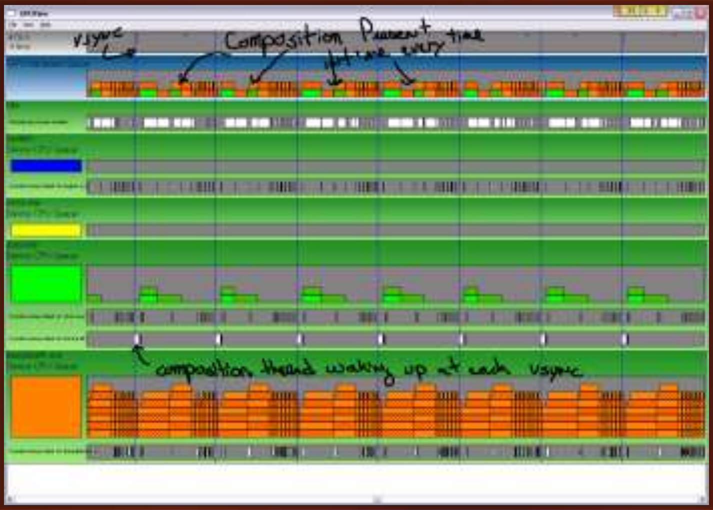


GPU

GPU Threading

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

GPU Context And Engine Queue



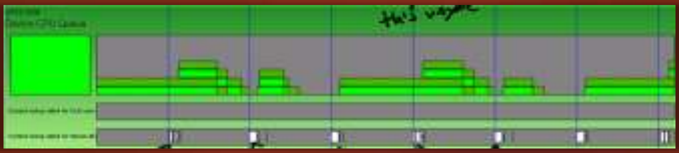
Context Queue

- The device queue (also referred to as the software queue or QueuePacket queue) represents the accumulation of work submitted by a process to a particular GPU Context
- A process can have multiple GPU contexts (ex: Multi D3D Device, LDA, Video…)

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Context Queue

- Queue is represented through time as a histogram
- On submission of work by the UMD the histogram grows up, on completion of work by the GPU the histogram goes down



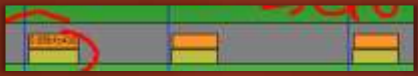
Queue Packet

- Various type of packet may be queued
  - Orange: MMIO flip
  - Diag-Cross Hatch: Present through DMA
  - Cross Hatch: Deferred Command
  - Light Pink: WaitForSynchronizationObject
  - Dark Pink: SignalSynchronizationObject

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

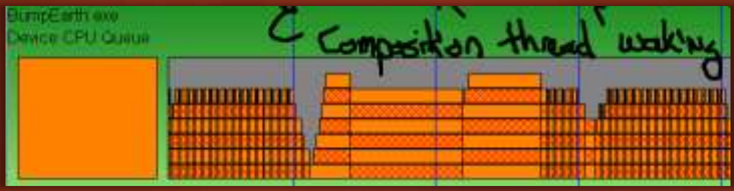
MMIO Flip

- Completion of the packet indicate the flip was pushed to the flip queue
- Flips are dequeued from the flip queue:
  - At VSYNC for synchronized flip
  - Immediately for immediate flip
- The content of the flip queue isn't currently displayed by GPUView
- Packet contain two pieces of information
  - VidPnSource : PrimaryAllocationHandle

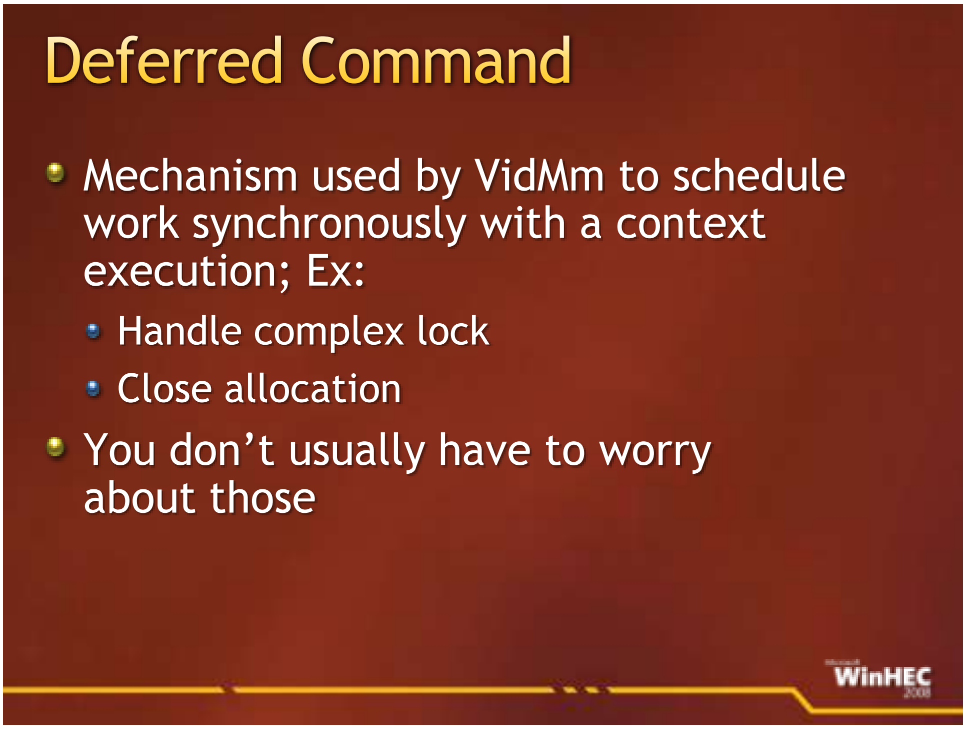


Present Through DMA

- Three types of present through DMA
  - DWM Redirection
  - Blit to back buffer
  - Immediate flip (through DMA)

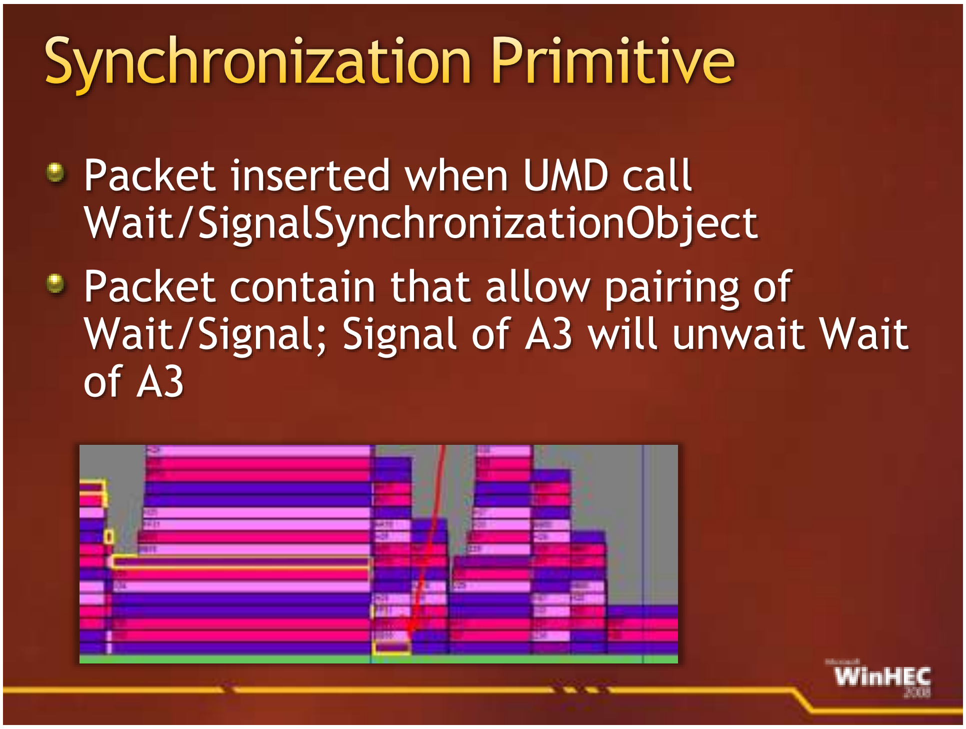


© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.



Deferred Command

- Mechanism used by VidMm to schedule work synchronously with a context execution; Ex:
  - Handle complex lock
  - Close allocation
- You don’t usually have to worry about those



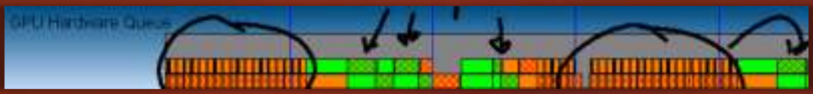
Synchronization Primitive

- Packet inserted when UMD call Wait/SignalSynchronizationObject
- Packet contain that allow pairing of Wait/Signal; Signal of A3 will unwait Wait of A3

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

GPU Engine Queue

- Also referred as hardware queue or DMAPacket queue
- Represent work (DMA Buffer) submitted to a particular GPU engine for execution
- Each exposed engine has exactly ONE engine queue



WinHEC 2008

DMA Packet

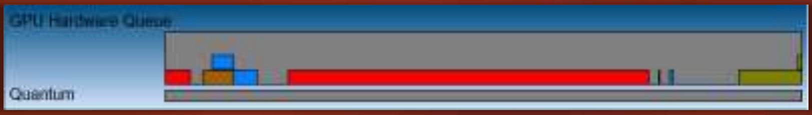
- Various type of DMA packet may be submitted to the GPU
  - Red: Paging
  - Black: Preemption
  - Diag-cross hatch: Present (through DMA)
  - Other: DMA payload from QueuePacket

WinHEC 2008

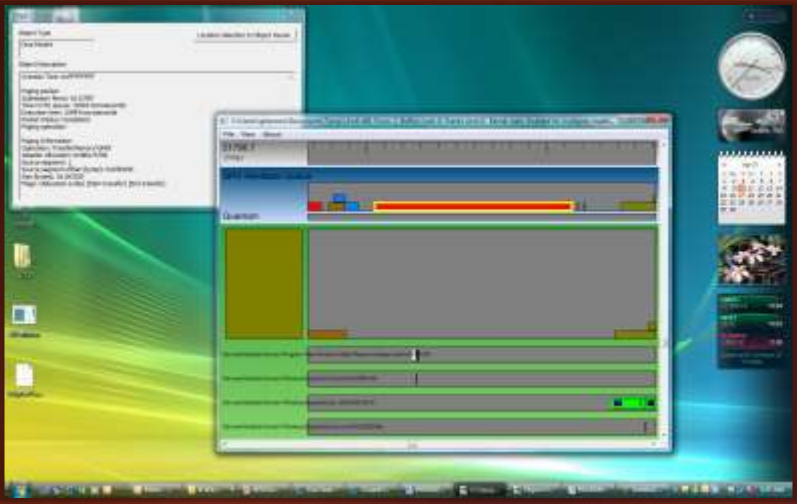
© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Paging Buffer

- Submitted as a result of a paging operation
- Cause of paging operation is usually the preparation of a DMA buffer; Look at the DMA packet following the paging operation
- Some exception
  - Pinning allocation
  - Evict allocation as a result of a DxLock
  - …



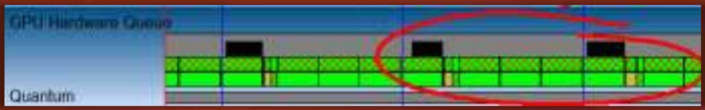
Paging Details



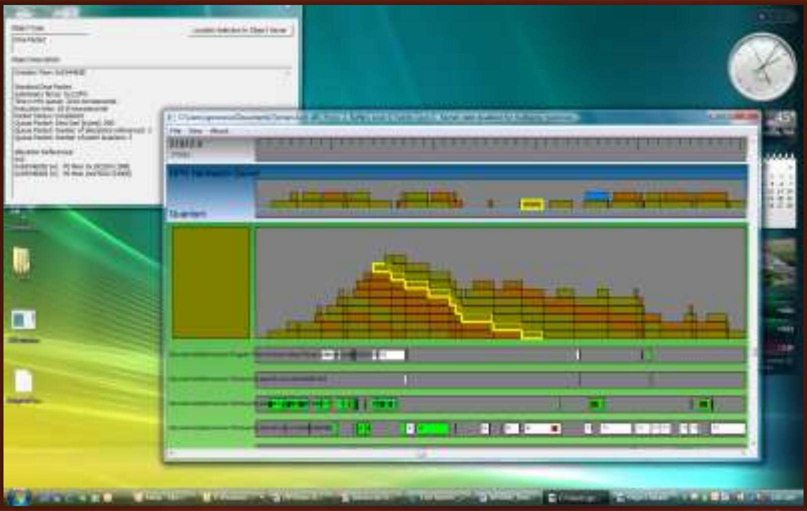
© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Preemption

- Represent request by the schedule to preempt the DMA packet currently in the engine queue
- Preemption request don’t have actual payload; Their completion indicate GPU acknowledgement of the preemption request



QueuePacket With DMA Payload

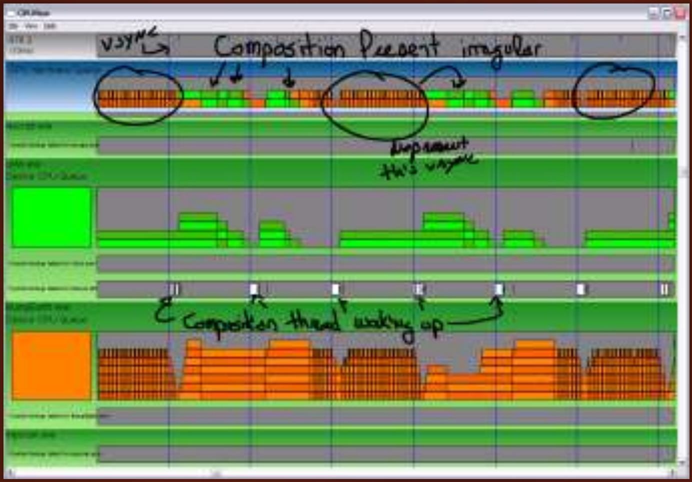


© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

demo

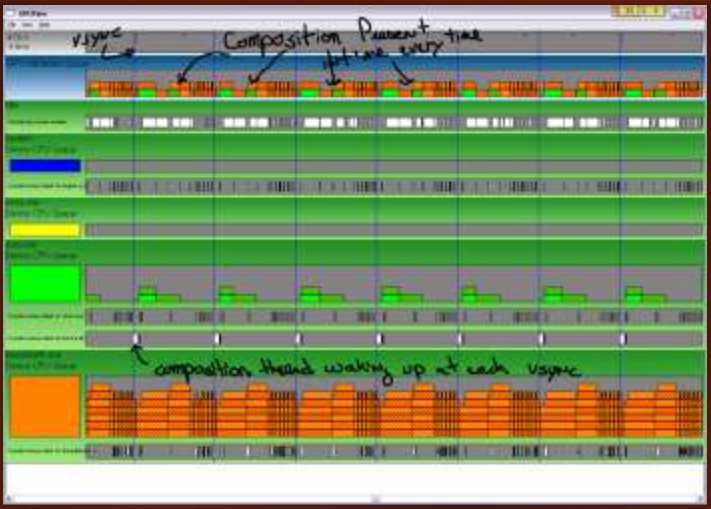
GPU Usage In GPUView

Ex: DWM Regular GPU Priority

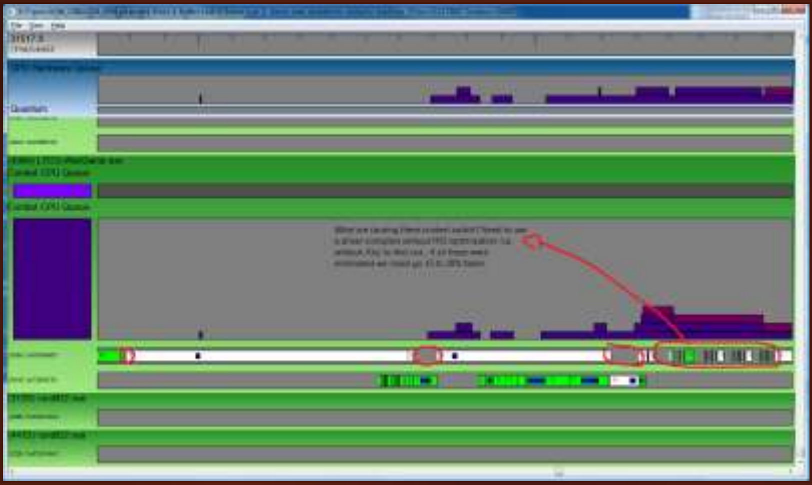


© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

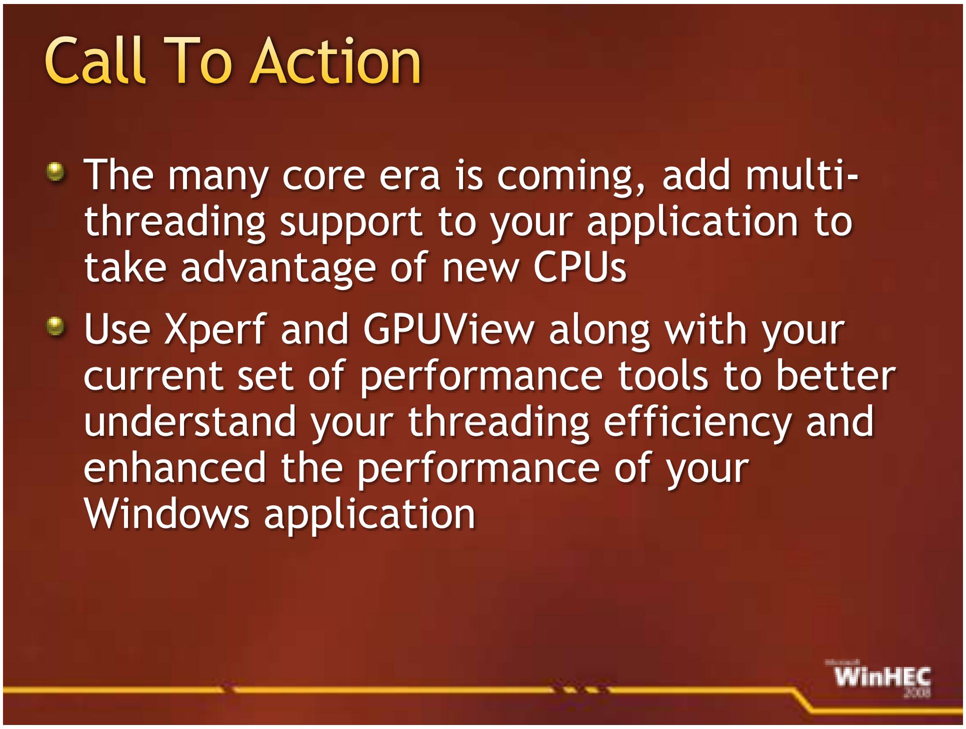
Ex: DWM High GPU Priority


vsync
Composition
Present picture every time
composition thread waking up at each vsync

Ex: Available Resources

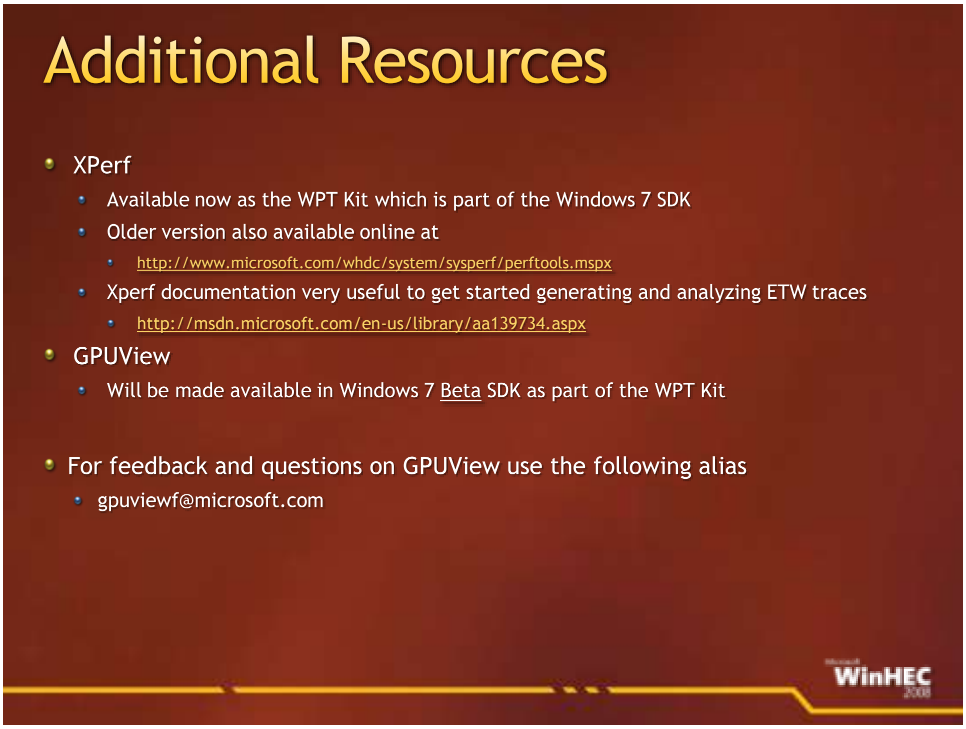


© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.



Call To Action

- The many core era is coming, add multi-threading support to your application to take advantage of new CPUs
- Use Xperf and GPUView along with your current set of performance tools to better understand your threading efficiency and enhanced the performance of your Windows application

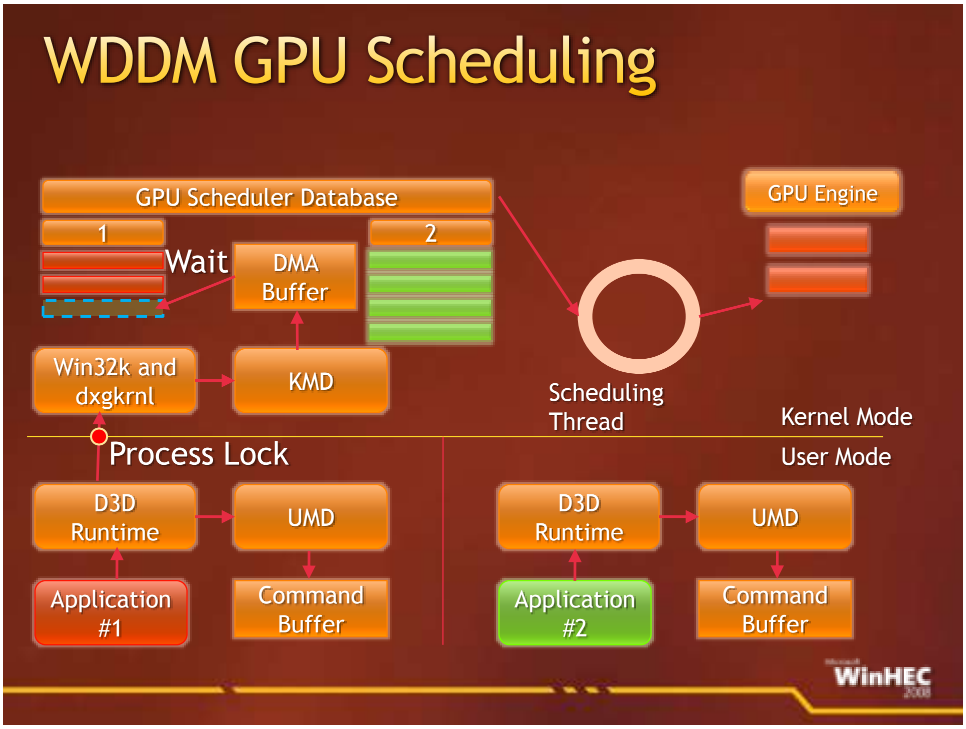


Additional Resources

- XPerf
  - Available now as the WPT Kit which is part of the Windows 7 SDK
  - Older version also available online at
    - http://www.microsoft.com/whdc/system/sysperf/perftools.mspx
  - Xperf documentation very useful to get started generating and analyzing ETW traces
    - http://msdn.microsoft.com/en-us/library/aa139734.aspx
- GPUView
  - Will be made available in Windows 7 Beta SDK as part of the WPT Kit
- For feedback and questions on GPUView use the following alias
  - gpuviewf@microsoft.com

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Appendix



WDDM GPU Scheduling

GPU Scheduler Database

1

2

Wait

DMA Buffer

Win32k and dxgkrnl

KMD

Scheduling Thread

GPU Engine

Kernel Mode

User Mode

Process Lock

D3D Runtime

UMD

Application #1

Command Buffer

D3D Runtime

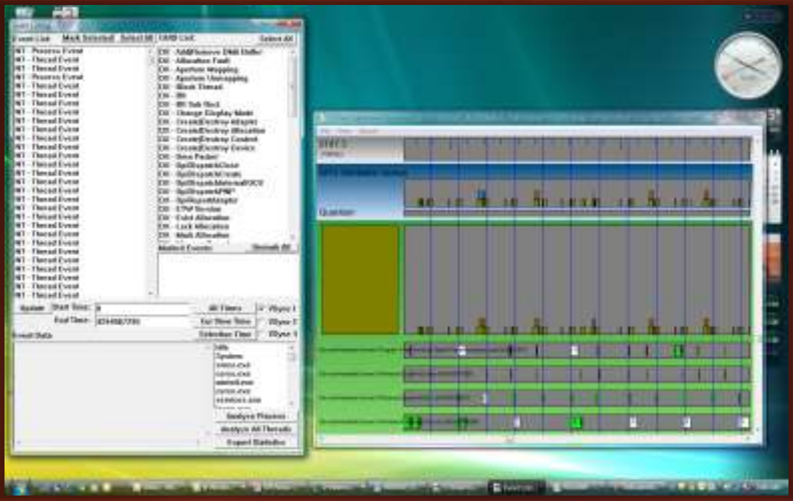
UMD

Application #2

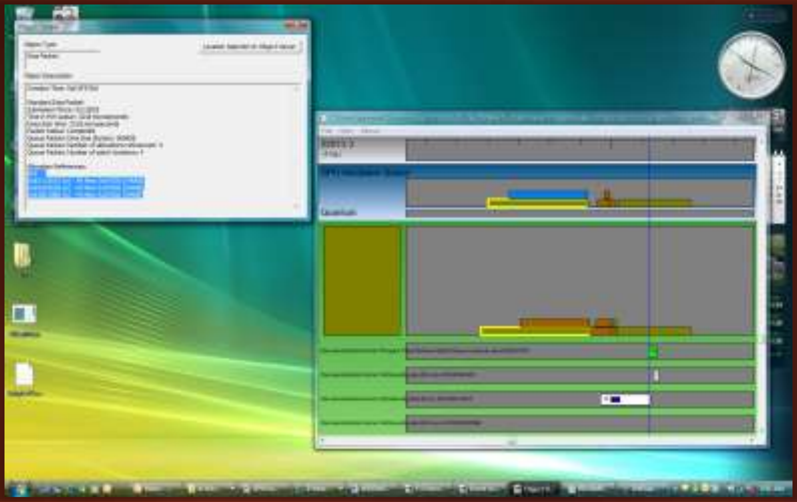
Command Buffer

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

VSync



AllocationList And Placement

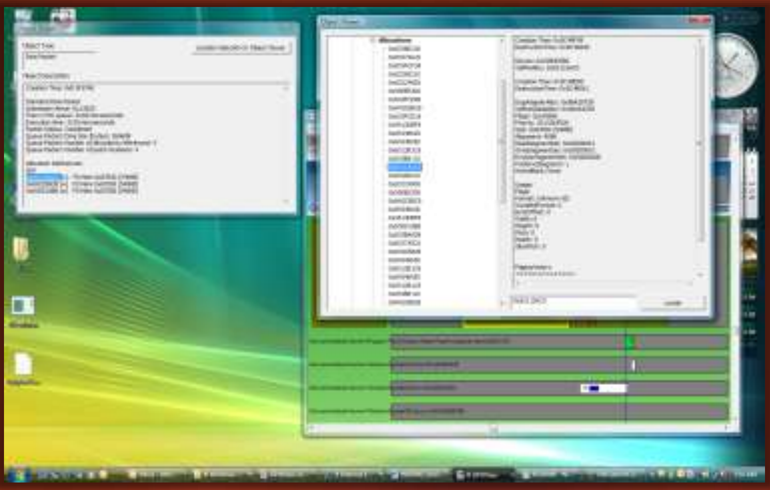


© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

AllocationList And Placement

- 0x83115AC0 (w) P0 Mem 0x87000 (540KB)
- Handle: Global identifier for the allocation
- Operation: (w) indicate a write operation. The default is read
- Relative priority: Where the allocation was placed relative to its priority
- Mem/App: Memory segment or Aperture Segment
- Size of allocation

More Details On Allocation



© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

Microsoft®

Your potential. Our passion.™

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.
The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.
MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.