

CPU Sets Sample

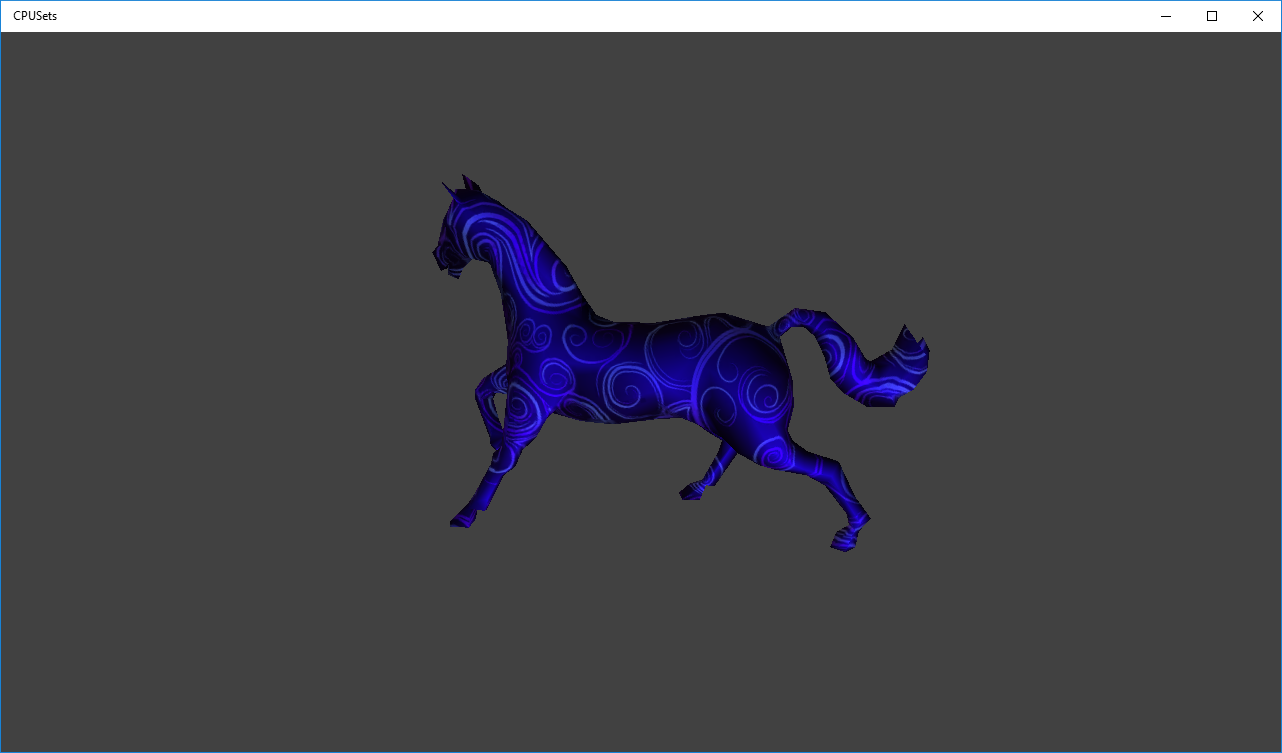
# *This sample is compatible with the Windows 10 April 2018 Update SDK (17134)*

# Description

This sample demonstrates using the CPU Sets APIs to ascertain information about the CPU cores on the system, then distributes a number of threads on those cores based on that information.

# Using the sample

There is minimal interaction with this sample. It simply displays a 3D model and plays a music track.



# Implementation notes

The interesting code in this sample is mainly in the **OrganizeCPUSets** function and the **SortThreads** function. The **OrganizeCPUSets** function demonstrates using **SYSTEM\_CPU\_SET\_INFORMATION** returned from **GetSystemCpuSetInformation** to determine the capabilities of the system. It uses the information to determine whether the system uses hyper threading, and organizes the CPU cores based on physical location. The **SortThreads** function uses that information to allocate threads on different cores with the intention of keeping particularly heavy threads on separate physical cores. It also demonstrates basic cross-thread communication between the **Render** loop and the **GeneratorThread**.

# Update history

* Initial release April 2016

# Privacy statement

When compiling and running a sample, the file name of the sample executable will be sent to Microsoft to help track sample usage. To opt-out of this data collection, you can remove the block of code in Main.cpp labeled “Sample Usage Telemetry”.

For more information about Microsoft’s privacy policies in general, see the [Microsoft Privacy Statement](https://privacy.microsoft.com/en-us/privacystatement/).