WINDOWS 7, WINDOWS 8, AND WINDOWS 8.1 APPRAISER TELEMETRY EVENTS AND FIELDS

Windows Customer Data Opt-in

The Windows Customer Data Opt-in (CDO) is an optional setting for enterprise-managed devices that instructs Windows to gather device-specific telemetry data. By configuring CDO, enterprises will be able to gather information from their Windows 7, Windows 8, or Windows 8.1 devices about apps, drivers, hardware configurations, and other engagement with the operating system to generate upgrade related insights.

Key: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\DataCollection
Value Name: CommercialDataOptIn
Value Type: DWORD
Value Data:
0 – Windows Customer Data Opt-in is disabled
1 – Windows Customer Data Opt-in is enabled

This document lists telemetry events, grouped by event area, and the fields within each event gathered by Windows 7, Windows 8, and Windows 8.1 through CDO for Windows Analytics: Upgrade Readiness. A brief description is provided for each field. Note that every event generated includes common data, which collects device data. This list of events and fields may be updated over time.
We are always striving to improve our documentation and welcome your feedback. You can provide feedback by contacting **telmhelp@microsoft.com**.

Upgrade Readiness

Through opting into CDO, you will have access to Upgrade Readiness, a solution that gives enterprises the tools to plan and manage the OS upgrade process end-to-end. Microsoft provides recommendations for resolving blocking issues, allowing you to streamline and accelerate Windows upgrades. Upgrade Readiness was developed following demand from customers looking for additional direction and detail about the Windows upgrade path, both from a time and cost perspective. We built Upgrade Readiness taking into account multiple channels of customer feedback, testing and Microsoft's experience upgrading millions of devices to Windows 10.

With Windows telemetry enabled through CDO, Upgrade Readiness collects and analyzes your data to identify device, application, and driver compatibility issues that can present a barrier to your upgrade process.

The Upgrade Readiness visual workflow steps you through the process to identify the devices that are ready to be upgraded.

With Upgrade Readiness you can:

- Get a detailed inventory of devices, applications, and drivers; drill down and search the data
- Decide which devices you want to use in a pilot and then export this list to your software deployment tool
- Prioritize the apps that are most important to the organization so you can start to resolve any blockers
- See which apps are used the most, grouped by the devices they're installed on
- Track validation progress and decisions
- Decide which devices are ready to be upgraded and then export this list to your software deployment tool

Note: Internet Explorer data collection (Microsoft.Windows.App.Browser and Microsoft.Web.Platform) is disabled by default and will only be enabled if the following registry value is configured. Additional information about this configuration can be found here.

Key: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\DataCollection
Value Name: IEDataOptIn
Value Type: DWORD
Value Data:

0 – Internet Explorer data collection is disabled
1 – Data collection is enabled for sites in the Intranet + Trusted + Local Zones
2 – Data collection is enabled for sites in the Internet + Restricted Zone
3 – Data collection is enabled for all sites

Events	Fields	Description
Common Data Events		This is common device data that is added to every event.
Common Data Envelope		
	Ver	Represents the major and minor version of the envelope.
	name	Represents the uniquely qualified name for the event.
	time	Represents the event date time in Coordinated Universal Time when the event was generated on the client. This should be in ISO 8601 format.
	popSample	Represents the effective sample rate for this event at the time it was generated by a client.
	epoch	Represents the epoch and seqNum fields, which help track how many events were fired and how many events were uploaded, and enables identification of data lost during upload and de-duplication of events on the ingress server.
	seqNum	Represents the sequence field used to track absolute order of uploaded events. It is an incrementing identifier for each event added to the upload queue. The Sequence helps track how many events were fired and how many events were uploaded and enables identification of data lost during upload and de-duplication of events on the ingress server.
	iKey	Represents an ID for applications or other logical groupings of events .
	flags	Represents a collection of bits that describe how the event should be processed by the Connected User Experience and Telemetry component pipeline. The lowest-order byte is the event persistence. The next byte is the event latency.
	os	Represents the operating system name.
	osVer	Represents the OS version, and its format is OS dependent.
	appId	Represents a unique identifier of the client application currently loaded in the process producing the event; and is used to group events together and understand usage pattern, errors by application.
	appVer	Represents the version number of the application. Used to understand errors by Version, Usage by Version across an App.
	cV	Represents the Correlation Vector: A single field for tracking partial order of related telemetry events across component boundaries.
	tags	Represents the pre-release build "flight ID"
Common Data – Telemetry Extension		
	stId	Represents the Scenario Entry Point ID. This is a unique GUID for each event in a diagnostic scenario. This used to be Scenario Trigger ID.
	aId	Represents the ETW ActivityId. Logged via TraceLogging or directly via ETW.
	raId	Represents the ETW Related ActivityId. Logged via TraceLogging or directly via ETW.
	Op	Represents the ETW Op Code.
	cat	Represents a bitmask of the ETW Keywords associated with the event.
	flags	Represents the bitmap that captures various Windows specific flags.
Common Data – Device Extension		
	device:localId	Represents a locally defined unique ID for the device, not the human readable device name. Most likely equal to the value stored at HKLM\Software\Microsoft\SQMClient\MachineId
	device:deviceClass	Represents the classification of the device, the device “family”. For example, Desktop, Server, or Mobile.
Common Data – User Extension		
	user:localId	Represents a unique user identity that is created locally and added by the client. This is not the user's account ID.
Common Data – OS Extension		
	os: expId	Represents the “experiment ID”. The standard for associating a flight, such as an OS flight (pre-release build), or an experiment, such as a web site UX experiment, with an event is to record the flight / experiment IDs in Part A of the common schema.
Consent UI Event		This User Account Control (UAC) telemetry point collects information on elevations that originate from low integrity levels. This occurs when a process running at low integrity level (IL) requires higher (administrator) privileges, and therefore requests for elevation via UAC (consent.exe). By better understanding the processes requesting these elevations, Microsoft can in turn improve the detection and handling of potentially malicious behavior in this path
	eventType	Represents the type of elevation: If it succeeded, was cancelled, or was auto-approved.
	splitToken	Represents the flag used to distinguish between Admin and Standard Users.

	friendlyName	Represents the name of the file requesting elevation from low IL.
	elevationReason	Represents the distinction between various elevation requests sources (appcompat, installer, COM, MSI and so on).
	exeName	Represents the name of the file requesting elevation from low IL.
	signatureState	Represents the state of the signature, if it signed, unsigned, OS signed and so on.
	publisherName	Represents the name of the publisher of the file requesting elevation from low IL.
	cmdLine	Represents the full command line arguments being used to elevate.
	Hash.Length	Represents the length of the hash of the file requesting elevation from low IL.
	Hash	Represents the hash of the file requesting elevation from low IL.
	HashAlgId	Represents the algorithm ID of the hash of the file requesting elevation from low IL.
	telemetryFlags	Represents the details about the elevation prompt for CEIP data.
	timeStamp	Represents the time stamp on the file requesting elevation
	fileVersionMS	Represents the major version of the file requesting elevation
	fileVersionLS	Represents the minor version of the file requesting elevation
Appraiser events		These events (Microsoft.Windows.Appraiser.* and Microsoft.Windows.Inventory.*) provide an inventory of the device along with an appraisal against that inventory for the purposes of understanding compatibility and upgrade issues. This device inventory gathers information such as applications, devices, drivers and other system info that we track to be able to assess the ability to upgrade. The events also include information about what we know about that object from a compatibility perspective (contained in the DataSource related events), and what our final decision is for that item on whether it can be upgraded (contained within the Decision related events). Majority of the events tend to follow a common pattern revolving around an ObjectType, such as an application or driver. There are StartSync events that are just indicators that we are starting a new collection on an ObjectType, Add events indicating we are adding data for that object type, Remove Events indicating we are deleting the data for that ObjectType and finally an EndSync event indicating we're finishing the collecting for that Object type.
Microsoft.Windows.Appraiser.General.		
InventoryApplicationAdd		This event represents the basic metadata about an application installed on the system.
	objectInstanceId	ProgramId (a hash of Name, Version, Publisher, and Language of an application used to identify it). Example: 00000144865763f3de24c2ae5a289fde6db300000904
	HiddenArp	Indicates whether a program hides itself from showing up in ARP. Example: TRUE
	InstallDate	The date the application was installed (a best guess based on folder creation date heuristics) Example: 4/12/2015 01:27:52
	InstallDateArpLastModified	The date of the registry ARP key for a given application. Hints at install date but not always accurate. Passed as an array. Example: 4/11/2015 00:00:00
	InstallDateFromLinkFile	The estimated date of install based on the links to the files. Passed as an array. Example: 4/8/2015 01:06:11
	InstallDateMsi	The install date if the application was installed via MSI. Passed as an array. Example: 4/11/2015 00:00:00
	Language	The language code of the program. Language codes can be found at http://support.microsoft.com/kb/221435 Example: 1033
	MsiPackageCode	A GUID that describes the MSI Package. Multiple 'Products' (apps) can make up an MsiPackage. Example: {1BCC5142-D98C-430B-B74A-484A0328A7CE}
	MsiProductCode	A GUID that describe the MSI Product. Example: {365812a8-44d6-422e-b737-d540451e5f4e}
	Name	The name of the application. Location pulled from depends on 'Source' field.
	OSVersionAtInstallTime	The four octets from the OS version at the time of the application's install.
	PackageFullName	The package full name for a Store application. Example: Microsoft.Hexic_1.2.0.36_x86__8wekyb3d8bbwe
	ProgramInstanceId	A hash of the file IDs in a program. Used to identify application install footprint. Example: 00002a54cb9c5bc6946b99d4180fec12d6c1103ad849
	Publisher	The Publisher of the application. Location pulled from depends on the 'Source' field. Example: Neudesic
	RootDirPath	The path to the root directory where the program was installed. Example: %ProgramFiles% (x86)\Neudesic\Azure Storage Explorer 6

	Source	Where the data for the application was found, such as Add/Remove Programs (ARP), MSI, AppxPackage, etc. Example: Msi
	Type	One of ("Application", "Hotfix", "BOE", "Service", "Unknown"). Application indicates Win32 or Appx app, Hotfix indicates app updates (KBs), BOE indicates it's an app with no ARP or MSI entry, Service indicates that it is a service. Application and BOE are the ones most likely seen. Example: Application
	Version	The version number of the program. Example: 6.00.0003
InventoryApplicationFileAdd		This event represents the basic metadata about a file on the system. The file must be part of an app and either have a block in the compatibility database or are part of an anti-virus program.
	objectInstanceId	LongPathHash: A hash of the full file path including the file name. Example: 00002e017145d5fedc3dd5dd4027b1da51d17ca2a0a3
	BinFileVersion	An attempt to clean up FileVersion at the client that tries to place the version into 4 octets. Example: 12.0.31101.0
	BinProductVersion	An attempt to clean up ProductVersion at the client that tries to place the version into 4 octets. Example: 12.0.31101.0
	BinaryType	One of ("UNINITIALIZED", "ZERO_BYTE", "DATA_ONLY", "DOS_MODULE", "NE16_MODULE", "PE32_UNKNOWN", "PE32_I386", "PE32_ARM", "PE64_UNKNOWN", "PE64_AMD64", "PE64_ARM64", "PE64_IA64", "PE32_CLR_32", "PE32_CLR_IL", "PE32_CLR_IL_PREFER32", "PE64_CLR_64"). Example: PE32_I386
	BoeProgramId	The ProgramId generated from the file metadata if the file is an orphan file (no ARP, MSI, etc. entry). BOE means Bag of Evidence.
	CompanyName	The company name of the vendor who developed this file. Example: Microsoft Corporation
	FileId	A hash that uniquely identifies a file. Example: 0000eef5472f6619824665a9c118cffea67b3727f0e1
	FileVersion	The File version field from the file metadata under Properties -> Details. Example: 12.0.31101.0 built by: REL
	LinkDate	The DateTime this file was linked on. Example: 11/1/2014 7:09:24 AM
	LowerCaseLongPath	The full file path of the executable on the machine this was file was inventoried on. Example: %ProgramFiles% (x86)\microsoft visual studio 12.0\common7\ide\devenv.exe
	Name	The name of the file that was inventoried. For example, excel.exe
	ProductName	The Product name field from the file metadata under Properties -> Details. Example: Microsoft® Visual Studio® 2013
	ProductVersion	The Product version field from the file metadata under Properties -> Details. Example: 12.0.31101.0
	ProgramId	A hash of Name, Version, Publisher, and Language of an application used to identify it. Example: 00004a73716911b8bb891ec1f536f2bf500b00000904
DecisionApplicationFileAdd		This event sends true/false compatibility decision data about a file to help keep Windows up to date.
	objectInstanceId	LongPathHash: A hash of the full file path including the file name. Example:
	BlockAlreadyInbox	Indicates that the uplevel runtime block on the file already existed on the current OS and is therefore not a regression. Example: FALSE
	BlockingApplication	Indicates if there are any application issues that interfere with upgrade due to the file in question. Example: FALSE
	DisplayGenericMessage	Indicates if there will be a generic message shown for this file. Example: FALSE
	HardBlock	The file is hardblocked in the SDB and can't run uplevel. Example: FALSE
	HasUxBlockOverride	The file has a block that is overridden by a tag in the SDB to have it not show up in reports or to the user (e.g. Intel CPL). Example: FALSE
	MigApplication	The file has a MigXML from the SDB associated with it that applies to the current upgrade mode. Example: FALSE
	MigRemoval	The file has a MigXML from the SDB that will cause the app to be removed on upgrade. Example: FALSE
	NeedsDismissAction	Indicates the file will be bubbled up to setup as a dismissible action. Example: FALSE
	NeedsInstallPostUpgradeData	Indicates that after upgrade, this file will have a post-upgrade notification to install a replacement for the application (requires a situation that the file must be uninstalled to upgrade). Example: FALSE
	NeedsNotifyPostUpgradeData	Indicates the file has a notification mig that should be surfaced in post-upgrade. Example: FALSE
	NeedsReinstallPostUpgradeData	Indicates that after upgrade, this file will have a post-upgrade notification to reinstall the app. Example: FALSE
	NeedsUninstallAction	The file must be uninstalled to upgrade. Example: FALSE
	SdbBlockUpgrade	The file is tagged as blocking upgrade in the SDB. Example: FALSE
	SdbBlockUpgradeCanReinstall	The file is tagged as blocking upgrade in the SDB but can be reinstalled after upgrade. Example: FALSE

	SdbBlockUpgradeUntilUpdate	The file is tagged as blocking upgrade in the SDB but if the app is updated the upgrade can proceed. Example: FALSE
	SdbReinstallUpgrade	The file is tagged as needing to be reinstalled after upgrade in the SDB (but not blocking upgrade). Example: FALSE
	SdbReinstallUpgradeWarn	The file is tagged as needing to be reinstalled after upgrade with a warning in the SDB (but not blocking upgrade). Example: FALSE
	SoftBlock	The file is softblocked in the SDB and has a warning uplevel. Example: FALSE
DatasourceApplicationFileAdd		This event represents the compatibility information (database entries, registered as anti-virus, predicted to be compatible) for a file.
	objectInstanceId	LongPathHash: A hash of the full file path including the file name. Example: 0000eb6aa77318dd7af6737658f6045a9ddd80339602
	AvDisplayName	The display name for the app if it is an AV. Example: System Center Endpoint Protection
	CompatModelIndex	The compatibility prediction for this file. Will always be an empty string.
	HasCitData	Whether or not the file is present in CIT data. Example: FALSE
	HasUpgradeExe	Whether or not the AV has an upgrade.exe. Example: TRUE
	IsAv	Whether or not the file is an AV reporting EXE. Example: TRUE
	ResolveAttempted	Will always be an empty string when sending telemetry.
	SdbEntries	An array of fields indicating the SDB entries that apply to this file.
	SdbEntries_item_MigShimCommand	The command line to be passed to the MigShim if one is applicable.
	SdbEntries_item_MigShimName	Example: DevenvDotnetCacheRebuildShim
	SdbEntries_item_MigXmlName	Example: MicrosoftForefrontEndpointProtection__4_6__PART
	SdbEntries_item_MigXmlType	Example: MIG_XML_TYPE_REMOVED
	SdbEntries_item_ReinstallUpgradeMessage	Example: Resource: 10022
	SdbEntries_item_SdbAppGuid	Example: {551f8360-14dd-4ea5-bd29-74b0c21abfde}
	SdbEntries_item_SdbAppName	Example: Visual Studio
	SdbEntries_item_SdbAppVendor	Example: Microsoft
	SdbEntries_item_SdbBlockType	Example: ReinstallAfterUpgradeInfo
	SdbEntries_item_SdbEntryGuid	Example: {84e92468-a463-4c02-93a6-20171694b8a8}
	SdbEntries_item_SdbUpgradeMode	Example: Swap
	SdbEntries_item_SdbUxBlocktypeOverride	Example: SDB_UX_BLOCKTYPE_OVERRIDE_MIG_FIXED
DataSourceMatchingInfoBlockAdd		This event represents the compatibility database information about any compatibility blocking entries hit on the system that are not directly related to specific applications or devices.
	objectInstanceId	Will always be "BlockingMatchingInfo."
	SdbEntries	An array of fields indicating the SDB entries that apply to this file.
	SdbEntries_item_SdbAppGuid	Example: {4cca1f6c-74f8-4bfd-9fb4-3d4b65f23f98}
	SdbEntries_item_SdbAppName	Example: Intel(R)DynamicPowerPerformanceManagement
	SdbEntries_item_SdbAppVendor	Example: Intel
	SdbEntries_item_SdbBlockType	Example: BlockUpgradeUntilUpdate
	SdbEntries_item_SdbEntryGuid	Example: {4be49993-88ec-4003-b9a6-9f8812e94c50}
	SdbEntries_item_SdbUpgradeMode	Example: Swap
	SdbEntries_item_SdbUxBlocktypeOverride	Example: SDB_UX_BLOCKTYPE_OVERRIDE_UPGRADE_UNTIL_UPDATE_BLOCK
DecisionMatchingInfoBlockAdd		This event represents the result of all compatibility decisions (true/false) about blocking entries hit on the system that are not keyed by applications or devices.
	objectInstanceId	Will always be "BlockingMatchingInfo"
	BlockingApplication	Indicates if there are any application issues that interfere with upgrade due to matching info blocks. Example: FALSE
	DisplayGenericMessage	Indicates if there will be a generic message shown for this block. Example: FALSE
	NeedsUninstallAction	Does the user need to take an action in setup due to a matching info block? Example: FALSE
	SdbBlockUpgrade	Indicates if a matching info block blocks upgrade. Example: FALSE
	SdbBlockUpgradeCanReinstall	Indicates if a matching info block blocks upgrade but has the can reinstall tag. Example: FALSE
	SdbBlockUpgradeUntilUpdate	Indicates if a matching info block blocks upgrade but has the until update tag. Example: FALSE
DataSourceMatchingInfoPassiveAdd		This event represents the compatibility database information about any compatibility non-blocking entries hit on the system that are not keyed by applications or devices.

	objectInstanceId	Will always be "PassiveMatchingInfo."
	SdbEntries	An array of fields indicating the SDB entries that apply to this file.
	SdbEntries_item_MigShimCommand	The command line to be passed to the MigShim if one is applicable.
	SdbEntries_item_MigShimName	Example: MigrateVCRuntimeShim
	SdbEntries_item_MigXmlName	Example: Intel_Rapid_Storage_Technolgy_Enterprise_Filter_Driver__3__PART
	SdbEntries_item_MigXmlType	Example: MIG_XML_TYPE_FIXED
	SdbEntries_item_SdbAppGuid	Example: {03760bce-35d7-47a3-b83b-de673fdb6ab4}
	SdbEntries_item_SdbAppName	Example: VC Runtime
	SdbEntries_item_SdbAppVendor	Example: Microsoft
	SdbEntries_item_SdbBlockType	Example: BlockUpgradeUntilUpdate
	SdbEntries_item_SdbEntryGuid	Example: {00b0c9b2-3f04-4795-a8ac-5b7bd5ea2ea8}
	SdbEntries_item_SdbUpgradeMode	Example: Swap
	SdbEntries_item_SdbUxBlocktypeOverride	Example: SDB_UX_BLOCKTYPE_OVERRIDE_MIG_FIXED
DecisionMatchingInfoPassiveAdd		This event represents the result of all compatibility decisions (true/false) about non-blocking entries hit on the system that are not keyed by applications or devices.
	objectInstanceId	Will always be "PassiveMatchingInfo"
	BlockingApplication	Indicates if there are any application issues that interfere with upgrade due to matching info blocks. Example: FALSE
	MigApplication	Indicates if there is a matching info with a mig for the current mode of upgrade. Example: FALSE
DataSourceMatchingInfoPostUpgradeAdd		This event represents the compatibility database information about any entries requiring reinstallation after upgrade hit on the system that are not keyed by applications or devices.
	objectInstanceId	Will always be "PostUpgradeMatchingInfo"
	SdbEntries	An array of fields indicating the SDB entries that apply to this file.
	SdbEntries_item_ReinstallUpgradeMessage	Example: Resource: 10022
	SdbEntries_item_SdbAppGuid	Example: {0ba2f09d-5288-45fa-be32-001857cc020f}
	SdbEntries_item_SdbAppName	Example: Virtual Machine Manager Self-Service Client
	SdbEntries_item_SdbAppVendor	Example: Microsoft Corporation
	SdbEntries_item_SdbBlockType	Example: ReinstallAfterUpgrade
	SdbEntries_item_SdbEntryGuid	Example: {2a1cc617-9ee0-4dff-b3c0-a09cfc13543a}
	SdbEntries_item_SdbUpgradeMode	Example: Swap
	SdbEntries_item_SdbUxBlocktypeOverride	Example: SDB_UX_BLOCKTYPE_OVERRIDE_REINSTALL_BLOCK
DecisionMatchingInfoPostUpgradeAdd		This event represents the result of all compatibility decisions (true/false) about entries requiring reinstallation after upgrade hit on the system that are not keyed by applications or devices.
	objectInstanceId	Will always be "PostUpgradeMatchingInfo"
	NeedsInstallPostUpgradeData	Example: FALSE
	NeedsNotifyPostUpgradeData	Example: FALSE
	NeedsReinstallPostUpgradeData	Example: FALSE
	SdbReinstallUpgrade	Example: TRUE
InventoryApplicationIeAddonAdd		This event contains the basic metadata about an Internet Explorer add-on installed on the system. Windows Compatibility also collects information on Internet Explorer add-ons, similar to the information collected for other applications. Internet Explorer add-ons are frequently used by enterprise customers, and can be problematic on some versions of the OS. The data gathered for Internet Explorer add-ons is used to help resolve issues that customers face as part of deployment and OS upgrade.
	objectInstanceId	Example: {BDEADEF5-C265-11D0-BCED-00A0C90AB50F}
	BinFileVersion	Example: 15.0.4625.1000
	FileBinProductVersion	Example: 15.0.4625.0
	FileBinaryType	Example: PE32_I386
	FileCompanyName	Example: Microsoft Corporation
	FileDescription	Example: Microsoft Office Just-In-Time Virtualization Interceptor
	FileId	Example: 0000c633db3fd3733674d7ffc16c69ef81f0e4969978
	FileLinkDate	Example: 5/20/2014 8:29:19 AM
	FileLowerCaseLongPath	Example: c:\\program files\\microsoft office 15\\root\\office15\\interceptor.dll
	FileName	Example: Interceptor.dll

	FileOsComponent	If the file is an OS component
	FilePeChecksum	Example: 0x1f699
	FilePeHeaderHash	Example: 010111e422259d3e46ad4c56fc9c2424f9de617e40a0
	FileProductName	Example: Microsoft Office
	FileProductVersion	Example: 15.0.4625.1000
	FileSize	Example: 0x1eaf8
	FileSizeOfImage	Example: 0x22000
	FileSwitchBackContext	Example: 0x0100000000000601
	FileVerLanguage	Example: 1033
	FileVersion	Example: 15.0.4625.1000
	Name	Example: Microsoft Silverlight
	Publisher	Example: Microsoft Corporation
	Type	Example: ActiveX
Microsoft.Windows.Appraiser.General.Device data		This data is gathered to understand what devices are being used on Windows PCs and whether these devices will still work after the upgrade to the new OS. Windows Compatibility collects this data to determine how many blocking issues exist for these devices that are used on PCs, to identify what problems need to be solved for users to upgrade successfully.
InventoryDeviceContainerAdd		Represents the basic metadata about a device container, such as a monitor or printer (as opposed to a PNP device).
	objectInstanceId	ContainerId. Example: {552dd320-0dae-2794-2b41-df42fee22488}
	Categories	A list of functional categories that the container belongs to. Comma separated. Example: communication.phone,storage
	DeviceDataId	It is a hash from DeviceModelId, manufacturer, model name, primary category, and categories. Example: {f80f0f7f-d602-5b93-f740-a3e30a9a7840}
	DeviceModelId	It is a hash of all of the child devices' hwid. Example: {697805c1-6c6f-306e-4393-78cc9430d314}
	DiscoveryMethod	No value most of time. Example: ""
	FriendlyName	The name of container. Example: "Windows Phone"
	IsActive	IsConnected OR the device is last-seen less than 14 days. Example: 1
	IsConnected	More than IsPresent. For physically attached device, IsConnected is the same as IsPresent. For wireless device, IsConnected has communication link while IsPresent may not. Example: 1
	IsMachineContainer	Whether the container is the root machine itself. Example: 0
	IsNetworked	Networked device, e.g., a network printer. Example: 0
	IsPaired	This is for AEP (Association End Point) device, which is a type of devices that needs association(pairing) and authentication before use. Example: 0
	Manufacturer	Manufacturer name. Example: Polycom, Inc.
	ModelId	GUID for model. Example: "{da09b4cb-2391-a76e-4bb3-5f0fdc02c278}"
	ModelName	Name for model. Example: Polycom CX300
	ModelNumber	Not common, but is the model number for the device if set. Example: CX300
	PrimaryCategory	Primary category. Example: communication.phone
InventoryDevicePnpAdd		Represents the basic metadata about a PNP device and its associated driver.
	objectInstanceId	The Device Instance ID of the device (uniquely identifies a device in the system). Example: pci\ven_8086&dev_0085&subsys_13118086&rev_34\4&2dded11c&0&00e1
	COMPID	A JSON array the provides the value and order of the compatible ID tree for the device. Orders are integers, COMPIDs look like "usb\class_03&subclass_01&prot_01"
	ContainerId	System-supplied GUID that uniquely groups the functional devices associated with a single-function or multifunction device installed in the computer. Example: {27db0821-3bf9-f71a-f96f-a53403857690}
	Excluded	Excluded from the inventory report. Example: Never
	HWID	A JSON array that provides the value and order of the HWID tree for the device. Orders are integers, HWIDs look like "hid\vid_049f&pid_000e&rev_0102&mi_01&col04"
	STACKID	A JSON array that provides the value and order of the STACKID tree for the device. Orders are integers, STACKIDs look like "\driver\hidusb"
	audioCaptureDriver	A value indicates that the device loaded an Audio capture driver: wdma_bt.inf:997b70458bdb4031:BthHfAud:6.3.9600.17397:bthhfenum\\bthhfpaudio

audioRenderDriver	A value indicates that the device loaded an Audio render driver. Example: wdma_bt.inf:997b70458bdb4031:BthHfAud:6.3.9600.17397:bthhfenum\\bthhfpaudio
class	The device setup class of the driver loaded for the device. Example: net
classGuid	The GUID for the device setup class. Example: {4d36e972-e325-11ce-bfc1-08002be10318}
description	The device description from the driver INF. Example: Marvell AVASTAR Wireless-AC Network Controller
deviceState	DeviceState is a bitmask of the following: DEVICE_IS_CONNECTED 0x0001 (currently only for container) DEVICE_IS_NETWORK_DEVICE 0x0002 (currently only for container) DEVICE_IS_PAIRED 0x0004 (currently only for container) DEVICE_IS_ACTIVE 0x0008 (currently never set) DEVICE_IS_MACHINE 0x0010 (currently only for container) DEVICE_IS_PRESENT 0x0020 (currently always set) DEVICE_IS_HIDDEN 0x0040 DEVICE_IS_PRINTER 0x0080 (currently only for container) DEVICE_IS_WIRELESS 0x0100 DEVICE_IS_WIRELESS_FAT 0x0200 The most common values are therefore: 32 (0x20)= device is present 96 (0x60)= device is present but hidden 288 (0x120)= device is a wireless device that is present
displayDxLevel	The supported d3d feature level. One of the values in https://msdn.microsoft.com/en-us/library/windows/desktop/ff476329(v=vs.85).aspx. Example: 45312
displayHybridSupport	Whether the GPU is integrated or discrete in the hybrid system. Example: Integrated
displayWddmVersion	Windows Display Driver Model version. Example: 1300
driverId	Used to join with InventoryDriverBinary (on its key). Example: 0000f768ad5256cc5ef4b070bb5b99d1742028ca2bd3
driverInBox	Is the driver an inbox driver? Example: 1
driverIsKernelMode	Is it kernel mode driver? Example: 1
driverPackageId	Used to join with InventoryDriverPackage (on its key). Example: 0000128aa9f80512fef91fd1bd7f8c2ab8fee9a07689
driverPackageStrongName	The immediate parent directory name in the Directory field of InventoryDriverPackage. Example: wmiacpi.inf_amd64_de775fdac0374d06
driverSigned	Signing. Example: 1
driverVerDate	The DriverTimestamp (from InventoryDriverBinary) converted to a Date format, not the date from DriverVer in the driver INF. Example: 12/13/2014
driverVerVersion	The DriverVersion (from InventoryDriverBinary), not the version from DriverVer in the driver INF. Example: 6.3.9600.17617
enumerator	The bus that enumerated the device. Example: pci
fileName	Driver file name. Example: wificlass.sys
infPath	INF file name (the name could be renamed by OS, such as oemXX.inf)
installState	Device installation state. One of these values: https://msdn.microsoft.com/en-us/library/windows/hardware/ff543130(v=vs.85).aspx. Example: 0
lowerClassFilters	Lower class filter drivers' IDs, comma separated. Example: 00001deece9fa1408e84e5e6a54bf95869cb9e721172,0000f7c911352df091e37b4fd5f9864a558053bf325e
lowerFilters	Lower filter drivers' IDs, comma separated. Example: 00006f14b11f10d09b4c67f7a7189375cefece51c54a
manufacturer	Manufacturer name. Example: Marvell Semiconductor. Inc.
matchingID	Represents the hardware ID or compatible ID that Windows uses to install a device instance. Example: pci\\ven_11ab&dev_2b38&subsys_045e0001
model	Model name. Example: Marvell AVASTAR Wireless-AC Network Controller
originalInf	Original inf name from driver store. Example: mrvlpcie8897.inf
parentId	Device instance id of the parent of the device. Example: pci\\ven_8086&dev_9c14&subsys_9c141414&rev_e4\\3&11583659&1&e0

	printerDriver	Example: Xerox Office XPS Color Class Driver V1.2
	printerIeee1284DeviceId	Example: MFG:Xerox;CMD:PCL, PJL, PostScript;MDL:WorkCentre 7855;CLS:Printer;DES:Xerox WorkCentre 7855;CID:XR_PS_Office_Color
	problemCode	One of the CM_PROB_Xxx problem codes that are defined in \public\shared\inc\Cfg.h. Example: 0
	provider	Driver provider company. Example: Microsoft Corporation
	service	The name of the service that is installed for the device. Example: wificlass
	subClass	The subclass of the device, currently this is only set for devices that load a native WLAN or Mobile Broadband driver. Example: net.wlan.native
	upperClassFilters	Upper class filter drivers' IDs, comma separated. Example: 0000044ebee0e8bdf295fd4cd4e9f98962a7254b6e8d
	upperFilters	Upper filter drivers' IDs, comma separated. Example: 000011bd6e7f93660e7046714c8d025e28e5cb540984
DatasourceDevicePnpAdd		Represents the compatibility information (database entries, is boot critical, has a driver up-level) for a PNP device.
	objectInstanceId	The Device Instance ID of the device (uniquely identifies a device in the system). Example: pci\ven_8086&dev_0085&subsys_13118086&rev_34\4&2dded11c&0&00e1
	ActiveNetworkConnection	Is the device an active network device? Example: TRUE
	CosDeviceRating	Enumeration indicating if there is a driver on the target operating system. Example: 80
	CosDeviceSolution	Enumeration indicating how a driver on the target operating system is available. Example: 12
	CosDeviceSolutionUrl	Empty string
	CosPopulatedFromId	The expected uplevel driver matching ID based on driver coverage data. Example: hid_device_upr:ff00-ffff
	IsBootCritical	Is the device boot critical? Example: TRUE
	SdbEntries	An array of fields indicating the SDB entries that apply to this device.
	SdbEntries_item_SdbAppGuid	Example: {0ba2f09d-5288-45fa-be32-001857cc020f}
	SdbEntries_item_SdbAppName	Example: Virtual Machine Manager Self-Service Client
	SdbEntries_item_SdbAppVendor	Example: Microsoft Corporation
	SdbEntries_item_SdbBlockType	Example: ReinstallAfterUpgrade
	SdbEntries_item_SdbEntryGuid	Example: {2a1cc617-9ee0-4dff-b3c0-a09cfc13543a}
	SdbEntries_item_SdbUpgradeMode	Example: Swap
	SdbEntries_item_SdbUxBlocktypeOverride	Example: SDB_UX_BLOCKTYPE_OVERRIDE_REINSTALL_BLOCK
	UplevelInboxDriver	Is there a driver uplevel for this device? Example: TRUE
	WuDriverCoverage	Is there a driver uplevel for this device according to Windows Update? Example: TRUE
DecisionDevicePnpAdd		Represents the result of all compatibility decisions (true/false) about a PNP device.
	objectInstanceId	The Device Instance ID of the device (uniquely identifies a device in the system). Example: pci\ven_8086&dev_0085&subsys_13118086&rev_34\4&2dded11c&0&00e1
	AssociatedDriverIsBlocked	Indicates if the driver associated with this PNP device is blocked. Example: FALSE
	BlockAssociatedDriver	Indicates if the driver associated with this PNP device should be blocked based on a driver block authored against this PNP device. Example: FALSE
	BlockUpgradeIfDriverBlocked	Indicates if the PNP device is both boot critical and does not have an uplevel driver inbox. Example: FALSE
	BlockUpgradeIfDriverBlockedAndOnlyActiveNetwork	If this PNP device's driver is found to be blocked, this device will cause an upgrade block if there are no other unblocked active network devices on the system. Example: FALSE
	BlockingDevice	Indicates if the PNP device is blocking upgrade. Example: FALSE
	DisplayGenericMessage	Indicates if a generic message would be presented during setup for this PNP device. Example: FALSE
	DriverAvailableInbox	Is there an inbox driver available uplevel for this PNP device? Example: FALSE
	DriverAvailableOnline	Is there a WU driver available uplevel for this PNP device? Example: FALSE
	DriverAvailableUplevel	Is there an uplevel driver on WU or inbox for this PNP device? Example: FALSE
	DriverBlockOverridden	Indicates if there is a driver block on the device that has been overrided. Example: FALSE
	NeedsDismissAction	Indicates if the user would need to dismiss something during setup for this device. Example: FALSE
	NotRegressed	Indicates if the device has a problem code on the source OS that is no better than the one it would have on the target OS. Example: FALSE
	SdbDeviceBlockUpgrade	Indicates if the PNP device has an SDB block that blocks upgrade for the current upgrade mode. Example: FALSE

	SdbDriverBlockOverridden	Indicates if the PNP device has an SDB block that blocks upgrade for the current upgrade mode, but that block has been overridden. Example: FALSE
Microsoft.Windows.Appraiser.General. Driver data		Windows Compatibility gathers this data to understand driver usage, to ascertain whether apps and devices will work correctly for users after they upgrade to the new OS. This data is used to initially determine how many blocking issues exist. Then the Windows Compatibility team works to implement as many improvements as possible--both in the OS as well as working with our partners to provide improvements. The goal is to ensure that users have a successful experience on their PC after upgrade.
DatasourceDriverPackageAdd		Represents the compatibility database information about driver packages.
	objectInstanceId	DriverPackageId, used for uniquely identifying a driver package on a system. Example: 0000ce5aae60f3a60a9b9a715a6cee2053e305e7869c
	SdbEntries	An array of fields indicating the SDB entries that apply to this driver package.
	SdbEntries_item_SdbAppGuid	Example: {5f29791d-ad69-40a4-9783-6edbdf66bd4b}
	SdbEntries_item_SdbAppName	Example: Microsoft PDF/XPS Printer
	SdbEntries_item_SdbAppVendor	Example: Microsoft
	SdbEntries_item_SdbBlockType	Example: BlockDriver
	SdbEntries_item_SdbEntryGuid	Example: {380213ca-97c8-4fdc-b194-b4f714006796}
	SdbEntries_item_SdbUpgradeMode	Example: Service
	SdbEntries_item_SdbUxBlocktypeOverride	Example: SDB_UX_BLOCKTYPE_OVERRIDE_NO_BLOCK
DecisionDriverPackageAdd		Represents the result of all compatibility decisions (true/false) about a driver package.
	objectInstanceId	DriverPackageId, used for uniquely identifying a driver package on a system. Example: 0000ce5aae60f3a60a9b9a715a6cee2053e305e7869c
	DriverBlockOverridden	Indicates if the driver package has an SDB block that blocks it from migrating, but that block has been overridden. Example: FALSE
	DriverIsDeviceBlocked	Indicates if the driver package was blocked because of a device block. Example: FALSE
	DriverIsDriverBlocked	Indicates that the driver package has a driver block from the SDB (or is class blocked) Example: FALSE
	DriverShouldNotMigrate	Indicates if the driver package should not be migrated during upgrade. Example: FALSE
	SdbDriverBlockOverridden	Indicates if the driver package has an SDB block that blocks it from migrating, but that block has been overridden. Example: FALSE
InventoryDriverBinaryAdd		Contains the basic metadata about driver binaries running on the system.
	objectInstanceId	Can be used to join with InventoryDevicePnp (on driverId, upperFilters, etc.). Example: 000038dbe54a022b6c73edbdb8bf5cba32a882d2df2a
	DriverCheckSum	Driver file checksum. Example: 242561
	DriverCompany	Company name. Example: Advanced Micro Devices, Inc.
	DriverName	Driver file name. Example: 1394ohci.sys
	DriverPackageId	Package id. Example: 0000ac7d4445e02036880297d09945b84b0e203af804
	DriverPackageStrongName	Driver package StrongName. Example: acpi.inf_amd64_b82068c7a43a101f
	DriverTimeStamp	The low 32 bits of the time stamp of the image. This represents the date and time the image was created by the linker. The value is represented in the number of seconds elapsed since midnight (00:00:00), January 1, 1970, Universal Coordinated Time, according to the system clock. Example: 1429917363

	DriverType	Bitfield of driver attributes: 1. define DRIVER_MAP_DRIVER_TYPE_PRINTER 0x0001 2. define DRIVER_MAP_DRIVER_TYPE_KERNEL 0x0002 3. define DRIVER_MAP_DRIVER_TYPE_USER 0x0004 4. define DRIVER_MAP_DRIVER_IS_SIGNED 0x0008 5. define DRIVER_MAP_DRIVER_IS_INBOX 0x0010 6. define DRIVER_MAP_DRIVER_IS_WINQUAL 0x0040 7. define DRIVER_MAP_DRIVER_IS_SELF_SIGNED 0x0020 8. define DRIVER_MAP_DRIVER_IS_CI_SIGNED 0x0080 9. define DRIVER_MAP_DRIVER_HAS_BOOT_SERVICE 0x0100 10. define DRIVER_MAP_DRIVER_TYPE_I386 0x10000 11. define DRIVER_MAP_DRIVER_TYPE_IA64 0x20000 12. define DRIVER_MAP_DRIVER_TYPE_AMD64 0x40000 13. define DRIVER_MAP_DRIVER_TYPE_ARM 0x100000 14. define DRIVER_MAP_DRIVER_TYPE_THUMB 0x200000 15. define DRIVER_MAP_DRIVER_TYPE_ARMNT 0x400000 16. define DRIVER_MAP_DRIVER_IS_TIME_STAMPED 0x800000 Example: 8650778
	DriverVersion	Version. Example: 10.0.10074.0
	ImageSize	File size. Example: 602112
	NdisVersion	Network Driver Interface Specification (NDIS) framework version. Example: 6.3
	Product	Product name that is obtained from file info. Example: QLogic 10 GigE
	ProductVersion	Driver version that is obtained from file info. Usually the same as DriverVersion. Example: 7.12.2.3
	WdfVersion	Windows Driver Framework (WDF) version. Example: 1.15
InventoryDriverPackageAdd		Represents the basic metadata about the driver packages installed on the system.
	objectInstanceId	Used to join with InventoryDevicePnp (on driverPackageId). Example: 0000ce5aae60f3a60a9b9a715a6cee2053e305e7869c
	Class	Same as in InventoryDevicePnp. Example: printer
	ClassGuid	Same as in InventoryDevicePnp. Example: {4d36e979-e325-11ce-bfc1-08002be10318}
	Date	The manufacturer's build date of the driver. Note this could be earlier than DriverTimeStamp in the InventoryDriverBinary. Example: 5/28/2012
	Directory	Package directory full path. Example: C:\Windows\System32\DriverStore\FileRepository\hdxmb3hd.inf_amd64_cabd3ee20f1ff1b9
	Inf	Same as InfPath in InventoryDevicePnp. Example: oem9.inf
	InfSections_item_InfCharacteristics	Network component characteristics. Values see: https://msdn.microsoft.com/en-us/library/windows/hardware/ff547832(v=vs.85).aspx. Example: 40
	InfSections_item_InfClass	Network component class. Example: nettrans
	InfSections_item_InfDisplayName	Name of a network component that displays in the property sheet dialog box for a network connection. Example: Internet Protocol Version 6 (TCP/IPv6)
	InfSections_item_InfDriverId	The driver id in InventoryDriverBinary. Example: 0000b7c8a1a4c6576910a14963528689c598f0dc4183
	InfSections_item_InfInterfaceId	Instance GUID of a network component. Example: {65060815-3298-4178-9575-582f793f2e1c}
	InfSections_item_InfName	String ID for the network component. Example: ms_tcpip6
	InfSections_item_InfService	Service name for the network component. Example: NdisCap
	OriginalInf	Same as OriginalInf in InventoryDevicePnp. Example: abc.inf
	Provider	Provider for the package. Example: Microsoft
	SubmissionId	Driver package submission ID. Example: 1633333
	Version	Version of the package. Example: 6.0.7503.1
Microsoft.Windows.Appraiser.General.System data		This lightweight data about the device is collected to understand whether the individual device meets the minimum requirements to upgrade to the next version of the OS. Telemetry gathers information such as the amount of memory, the processor, and the BIOS. This data is used to determine thing such as whether there are memory constraints, or whether the device speed is too low for the upgrade OS.
InventorySystemMachineAdd		Represents the basic metadata about the currently running operating system on the system.
	objectInstanceId	Will always be "Machine"

	osArchitecture	Architecture for currently running OS from GetSystemInfo. 0 = x86 9 = AMD64 Example: 9
	osBuild	Build number from RtlGetVersion. Example: 7601
	osLangId	Return from GetSystemDefaultLangID. Example: 1033
	osMajor	OS major version from RtlGetVersion. Example: 6
	osMinor	OS minor version from RtlGetVersion. Example: 1
	osSpMajor	Service pack major version from RtlGetVersion. Example: 1
	osSpMinor	Service pack minor version from RtlGetVersion. Example: 0
SystemMemoryAdd		Represents the amount of memory on the system and if it meets requirements.
	objectInstanceId	Will always be "Memory"
	Blocking	Is the machine blocked from upgrade due to memory restrictions? Example: FALSE
	MemoryRequirementViolated	Indicates if a memory requirement was violated. Example: FALSE
	pageFile	The current committed memory limit for the system or the current process, whichever is smaller, in bytes. Example: 19650113536
	ram	Amount of RAM on the system. Example: 17099976704
	ramKB	ram/1024. Example: 16777216
	virtual	The size of the user-mode portion of the virtual address space of the calling process, in bytes. Example: 140737488224256
	virtualKB	virtual/1024. Example: 137438953344
InventorySystemProcessorAdd		Represents the basic metadata about the processor in the system.
	objectInstanceId	Will always be "Processor"
	architecture	Processor architecture. Example: x64
	caption	Processor caption. Example: Intel64 Family 6 Model 30 Stepping 5
	family	Processor family. Example: 198
	level	Number of cache levels. Example: 6
	level2CacheSize	Processor L2 cache size. Example: 1024
	level2CacheSpeed	Processor L2 cache speed. Example: 3647205776
	mhz	Processor speed in MHz. Example: 2799
	name	Processor name. Example: Intel(R) Core(TM) i5 CPU 750 @ 2.67GHz
	processorId	Processor ID. Example: BFEBFBFF000106E5
	stepping	Chip revision. Example: 4
	type	Processor type. Example: 3
	vendor	Processor vendor. Example: GenuineIntel
InventorySystemBiosAdd		Represents the basic metadata about the BIOS that is used to understand if it has a compatibility block.
	objectInstanceId	Will always be "Bios"
	biosDate	Release date of the Windows BIOS in the Coordinated Universal Time (UTC) format of YYYYMMDDHHMMSS.MMMMMM(+-)OOO from Win32_BIOS. Example: 20100607000000.000000+000
	biosName	Name field from Win32_BIOS. Example: Ver 1.00PARTTBLx
	manufacturer	Manufacturer field from Win32_ComputerSystem. Example: LENOVO
	model	Model field from Win32_ComputerSystem. Example: 2537J35
	AppraiserVersion	The version of the Appraiser file that is generating the events.
DatasourceSystemBiosAdd		Represents the compatibility database information about the BIOS.
	objectInstanceId	Will always be "Bios"
	SdbEntries	An array of fields indicating the SDB entries that apply to this BIOS.
	SdbEntries_item_SdbBlockType	Example: BiosBlock
	SdbEntries_item_SdbEntryGuid	Example: {b77118fd-0d87-4f63-a836-d5c6bd8eed4c}
	SdbEntries_item_SdbUpgradeMode	Example: Swap
DecisionSystemBiosAdd		Represents the result of all compatibility decisions (true/false) about the BIOS.
	objectInstanceId	Will always be "Bios"
	Blocking	Indicates if the device is blocked from upgrade due to a BIOS block. Example: FALSE
	HasBiosBlock	Does the device have a BIOS block? Example: FALSE

DecisionSystemProcessorAdd		Defunct data, always reads false.
	objectInstanceId	Will always be "Processor"
	Blocking	Will always be FALSE
	ProcessorRequirementViolated	Will always be FALSE
SystemProcessorCompareExchangeAdd		Represents if the system supports the CompareExchange128 CPU requirement.
	objectInstanceId	Will always be "CompareExchange128"
	Blocking	Indicates if upgrade is blocked due to this CPU check. Example: FALSE
	CompareExchange128Support	Indicates if the CPU supports CompareExchange128. Example: TRUE
SystemProcessorLahfSahfAdd		Represents if the system supports the LahfSahf CPU requirement.
	objectInstanceId	Will always be "LahfSahf"
	Blocking	Indicates if upgrade is blocked due to this CPU check. Example: FALSE
	LahfSahfSupport	Indicates if the CPU supports LAHF/SAHF. Example: TRUE
SystemProcessorNxAdd		Represents if the system supports the NX CPU requirement.
	objectInstanceId	Will always be "NX"
	Blocking	Indicates if upgrade is blocked due to this CPU check. Example: FALSE
	NXDriverResult	Provides the result of the driver used to check for NX support when not deterministic. Example: NotApplicable
	NXProcessorSupport	Indicates if the CPU supports NX. Example: TRUE
SystemProcessorPrefetchWAdd		Represents if the system supports the PrefetchW CPU requirement.
	objectInstanceId	Will always be "PrefetchW"
	Blocking	Indicates if upgrade is blocked due to this CPU check. Example: FALSE
	PrefetchWSupport	Indicates if the CPU supports PrefetchW. Example: TRUE
SystemProcessorSse2Add		Represents if the system supports the SSE2 CPU requirement.
	objectInstanceId	Will always be "SSE2"
	Blocking	Indicates if upgrade is blocked due to this CPU check. Example: FALSE
	SSE2ProcessorSupport	Indicates if the CPU supports SSE2. Example: TRUE
SystemTouchAdd		Represents if the system supports touch.
	objectInstanceId	Will always be "Touch"
	IntegratedTouchDigitizerPresent	Indicates if there is an integrated touch digitizer present. Example: FALSE
	MaximumTouches	Maximum number of touch points supported by the device hardware. Example: 1
SystemWimAdd		Represents if the current operating system is running from a compressed Windows Imaging Format (WIM) file.
	objectInstanceId	Will always be "Wim"
	IsWimBoot	Indicates if the current operating system is running from a compressed Windows Imaging Format (WIM) file. Example: FALSE
	RegistryWimBootValue	Raw value from registry used to indicate if system is running from a WIM. Example: 0
SystemWindowsActivationStatusAdd		Represents if the current operating system is activated.
	objectInstanceId	Will always be "WindowsActivationStatus"
	WindowsIsLicensedApiValue	Result from API used to indicate if operating system is activated. Example: TRUE
	WindowsNotActivatedDecision	Indicates if the current operating system is activated. Example: FALSE
SystemWlanAdd		Represents if the system has WLAN and if the WLAN runs using an emulated driver that would block the upgrade.
	objectInstanceId	Will always be "Wlan"
	Blocking	Indicates if upgrade is blocked due to an emulated WLAN driver. Example: FALSE
	HasWlanBlock	Indicates if upgrade is blocked due to an emulated WLAN driver. Example: FALSE
	WlanEmulatedDriver	Indicates if the system has an emulated WLAN driver. Example: FALSE
	WlanExists	Indicates if the device supports WLAN at all. Example: TRUE
	WlanModulePresent	Indicates if any WLAN modules are present. Example: TRUE
	WlanNativeDriver	Indicates if the system has a non-emulated WLAN driver. Example: TRUE
Microsoft.Windows.Appraiser.General.Miscellaneous		These events provide data about the user experience.
InventoryMiscellaneous		Lists information about the device's OS version, updateability, BIOS, battery capacity, location, power policy, manufacturer, etc.
	AUElevateNonAdmins	Tells us if the machine is configured for a non-admin to be able to install updates. 0 = No and 1 = Yes

AUIncludeRecommended	Tells us if the machine has been opted-in to install Recommended updates that same way as Important updates. 0 = No and 1 = yes
AUOptions	Tells us how the user has configured AU
ActiveSession	is there an active session at the time of the scan?
BaseBoardManufacturer	Baseboard manufacturer from HKLM\HARDWARE\\DESCRIPTION\\System\\BIOS\BaseBoardManufacturer
BaseBoardProduct	Baseboard product from HKLM\HARDWARE\\DESCRIPTION\\System\\BIOS\BaseBoardProduct
BaseBoardVersion	Baseboard version from HKLM\HARDWARE\\DESCRIPTION\\System\\BIOS\BaseBoardVersion
BatteryDesignedCapacity##	Original capacity of battery ## (00, 01, etc.) when fully charged, in milliwatt hours.
BatteryFullChargedCapacity##	Current capacity of battery ## (00, 01, etc.) when fully charged, in milliwatt hours.
BrowserChoiceEnabled	Is Browser Choice enabled feature configured on the machine? 0 = No and 1 = Yes From HKLM\SOFTWARE\\BrowserChoice\Enable
CbsLogSizeBytes	Example: 1928984759
CensusId	Unique ID created for WICABot which is a XOR of WER ID + SQM ID + WU ID. This has less duplication rate than SQM ID but does not persist over upgrade.
CrashDumpEnabled	Are crash dumps enabled?
CurrentPowerPolicy##	Currently active power policy. Should always be CurrentPowerPolicy00.
GeoId	Geographic location of the machine (as country code)
HasPaperLicense	Example : false
IeEnterpriseModeEnable	Is EMIE enabled? From HKLM\\SOFTWARE\\Policies\\Microsoft\\Internet Explorer\\Main\\EnterpriseMode
IeEnterpriseModeSiteList	EMIE site list from HKLM\\SOFTWARE\\Policies\\Microsoft\\Internet Explorer\\Main\\EnterpriseMode
IsAdmin	Is the currently logged in user an admin?
IsCensusDisabled	Example : 0
MrtId	MSRT [Malicious Software Removal Tool] ID of the machine from HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\RemovalTools\MRT
MrtVersion	MSRT version from HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\RemovalTools\MRT
OemManufacturerDate	Gives file time from %windir%\\csup.txt as the oem manufacturer date. Note that this is the date when the OS image was created. All machines carrying that image will have the same date.
OfficeReliabilityMID	Example: /OQOsgf0KEi8NCVq8COCgQ==
PIRWake	How many users have PIR wake disabled on Surface hub ?
PreviousUpgradeBuild	If the machine was upgraded, what was the previous OS?
SMBIOSUUID	SMBIOS UUID from the system BIOS Tables
SupportedGraphicsDXVersion	DirectX version supported by graphics card
SystemManufacturer	System manufacturer based on HKLM\HARDWARE\\DESCRIPTION\\System\\BIOS\SystemManufacturer
SystemProductName	System product name based on HKLM\HARDWARE\\DESCRIPTION\\System\\BIOS\SystemProductName
SystemSKU	System SKU based on HKLM\HARDWARE\\DESCRIPTION\\System\\BIOS\SystemSKU
UpgradeExperienceIndicatorsWritten	Example: true
VirtualMachineID	ID for correlation between HyperV guest and host
VolumeLicense	Is machine volume licensed?
WerDisabled	Is WER disabled?
WerId	WER ID, also called machine ID
WerPolicyDisabled	Is WER disabled via group policy?
WerService	WER Service State
WinSATCpuScore	Example: 7.4
WinSATDedicatedVideoMemory	Example: 0
WinSATDiskScore	Example: 8.15
WinSATGamingScore	Example: 5.4
WinSATGraphicsScore	Example: 5.9
WinSATMemoryScore	Example: 7.9
WinSATPrimaryVideoAdapter	Example: Intel(R) HD Graphics Family
WinSATVideoMemoryBandwidth	Example: 5762.01000
WinSxSSizeBytes	Example: 13011583208

	WindowsGenuineState	Windows genuine state in string form
	WinsatCpuUPExpressCompressThroughput	Compression throughput when running UPExpress
	WinsatDedicatedSystemMemory	Amount of system memory that is not available to the video adapter
	WinsatDedicatedVideoMemory	Amount of video memory not shared with the CPU in Bytes.
	WinsatPrimaryAdapterString	String describing primary video adapter
	WinsatSharedVideoMemory	Amount of video memory that is shared with the CPU in Bytes.
	WinsatTotalMonitorPixels	Total number of pixels (horizontal resolution * vertical resolution)
	WinsatVideoMemoryBandwidth	Video memory bandwidth in MB/s * 1000
	WinsatVideoMemorySize	Size of video memory in Bytes.
InventoryMiscellaneousAntispywareInformation		Lists information about the name and state of the Antispyware app installed on this device.
	Key	InstanceGUID from WMI query
	DisplayName	DisplayName from WMI query
	ProductState	ProductState from WMI query
InventoryMiscellaneousAntivirusInformation		Lists information about the name and state of the Antispyware app installed on this device.
	Key	InstanceGUID
	DisplayName	DisplayName
	ProductState	ProductState
InventoryMiscellaneousBrowserStartupSettings		Lists information about default browser name and startup pages.
	Key	Example: IEUserBrowserStartupSettings1471681315
	Browser	Example: IE
	Hash	Example: 1471681315
	SpecifiedPageUrls	Example: Not Applicable
	StandardUserHash	Example: 1471681315
	Value	Example: ShowHomePages
InventoryMiscellaneousCITModuleLoaded		Lists relevant details about the .NET library installed on the machine.
	Key	Example: Microsoft.NET/Framework/v1.1.4322/mscorsvr.dll|%SystemRoot%\System32\rundll32.exe
	BinaryPath	Example: %SystemRoot%\System32\rundll32.exe
	DataNamespace	Example: Module
	FileID	Example: 00008939cf35447b22dd2c6e6f443446acc1bf986d58
	ModuleBinFileVersion	Example: 1.1.4322.573
	ModulePath	Example: Microsoft.NET/Framework/v1.1.4322/mscorsvr.dll
	ProgramID	Example: 0000f519feec486de87ed73cb92d3cac802400000000
	TimeLoaded	Example: 130754960226340393
InventoryMiscellaneousChromeApp		Lists the Chrome apps installed on the system for all users. To assess the compatibility status of the apps.
	Key	{AppId}_{StandardUserHash}
	AppId	Chrome app ID, seem to be all letters
	OfflineEnabled	Is app available offline?
	ProfileHash	Hash of the user profile name
	StandardUserHash	Standardized user hash from the SID of the user.
	Version	Chrome app version
InventoryMiscellaneousChromeRlz		Represents information about how Chrome was installed.
	Key	{RlzString}#{StandardUserHash}
	Rlz	The RlzString
	StandardUserHash	Standardized user hash from the SID of the user.
	UserHash	Hash of the user profile name
InventoryMiscellaneousCpuid		Lists information about the device's CPU and virtualization capabilities.
	Key	Always "CpuIdData"
	CpuFamily	CPU family
	CpuModel	CPU model number
	CpuStepping	CPU stepping level
	IsVirtualizationSupported	Is virtualization supported on the device?
	IsHypervisorPresent	Whether or not a hypervisor is present. The hypervisor may or may not support the Hv#1 interface.

	IsIommuSupported	Is IOMMU (input/output memory management unit) supported. Will always return unknown on Win7.
	IsSLATSupported	IS SLAT (second level address translation) supported on the device?
InventoryMiscellaneousDiskInfo		Listing of disks on the device using the Win32_LogicalDisk WMI class.
	Key	DiskInfo + 2-digit running counter
	DriveType	Type of drive:
	FreeSpace	Free space on the disk
	HiberFileSize	Size of hiberfile on the disk
	PageFileSize	Size of pagefile on the disk
	Size	Size of the disk
InventoryMiscellaneousDiskPartitionInfo		Listing of disk partitions with data coming from WMI Win32_DiskPartition.
	Key	DiskPartitionInfo + 2-digit running counter
	DeviceId	Disk and partition numbers
	DriveLetters	Drive letter for the partition
	IsBootPartition	Is the partition a boot partition? Note that the boot partition is not necessarily where the OS is stored.
	IsPrimaryPartition	Is the partition a primary partition?
	Size	Size of the partition in bytes
	StartingOffset	Position on the disk of the partition in bytes
InventoryMiscellaneousMarkerCompatMarkers		Lists the markers used to determine the device compatibility of newer version of Windows.
	Key	Example: RAV
	IndicatorValue	Example: 0
	Key	Example: Free
	IndicatorValue	Example: It16
InventoryMiscellaneousMonitorData		Represents data about monitor resolutions as provided by Win32_VideoController WMI call. There will be two entries per monitor -- one providing the current resolution and one providing the max resolution. The two entries have different keys of PrimaryMonitor## and MaxMonitorResolution##.
	Key	(PrimaryMonitor OR MaxMonitorResolution) + 2-digit running counter
	HorizontalResolution	Horizontal resolution in pixels
	VerticalResolution	Vertical resolution in pixels
InventoryMiscellaneousPhysicalDiskInfo		Represents physical disk information from Win32_DiskDrive.
	Key	PhysicalDiskInfo + 2-digit running counter
	BusType	Indicates if machine has a hard drive, SSD, fast SSD (NVMe-attached), or a slow solid state storage, such as SD/MMC.
	BytesPerSector	Example: 512
	DeviceId	Unique identifier of the disk drive with other devices on the system.
	Index	Physical drive number of the given drive. A value of 0xFF indicates that the given drive does not map to a physical drive.
	MediaType	If the drive is a HDD or SSD (introduced in 1602)
	NumPartitions	Number of partitions on this physical disk drive that are recognized by the operating system.
	SerialNumber	Physical disk drive serial number
	Size	Size of the disk drive.
InventoryMiscellaneousServices		Represents a list of services present on the device.
	Key	Service name
	DisplayName	Service display name
	PathName	Service path
	StartMode	Service start mode
	State	Service current state
InventoryMiscellaneousSetupBootedFromAuditMode		Checks to see if the OS is currently running in Audit mode. This duplicates the check that is done as part of Setup to provide info on how many machines would hit it.
	Key	Always "BootedFromAuditMode"
	hostIsBootedFromAuditMode	Is the host booted into audit mode?
InventoryMiscellaneousSetupBootedFromVHD		Checks to see if the OS is currently booted from a VHD (does not mean this is a VM). This duplicates the check that is done as part of Setup to provide info on how many machines would hit it. Note: this event does not appear to get sent.

	Key	Always "BootedFromVHD"
	BootedFromVHD	Is the host booted into a VHD?
InventoryMiscellaneousSetupPendingFirmwareUpdateWithPower		Detects whether firmware update is pending because of power. This duplicates the check that is done as part of Setup to provide info on how many machines would hit it.
	Key	Always "PendingFirmwareUpdateWithPower"
	Pending	If a firmware update is pending because of power
InventoryMiscellaneousUser		Lists information about user defaults.
	Key	{Field}_{StandardUserHash}
	Exists	If the value exists
	OriginalName	The name of the field before the user hash was appended
	StandardUserHash	Standardized user hash from the SID of the user
	UserId	Hash of the either the SID or the user profile, depending on where the data point is collected. Use StandardUserHash to have a uniform view.
	Value	The value
	AdvertisingID	AdvertisingID is an ID generated by the browser when certain advertising sites want a tracking ID
	ChromeUserBrowserHomepage	Homepages for each Chrome user
	ChromeUserBrowserSearchSettings	Search engine settings for each Chrome user
	ChromeUserBrowserStartupSettings	What happens on startup: ShowBlankPage, ShowHomepages, ShowLastSession, ShowSpecifiedPages, UnexpectedSetting
	DVDTelemetrySessionStartDate	Start Date for a DVD session
	FirefoxUserBrowserHomepage	Homepages for each Firefox user
	FirefoxUserBrowserSearchSettings	Search engine settings for each Firefox user
	IEUserBrowserHomepage	Homepages for each IE user
	IEUserBrowserSearchSettings	Search engine settings for each IE user
	IeClearableListDataUserFilter	User Filter for IE Data User
	IeIndexId	Example:KRO7aRWaw6ZOWqAufK13+eDftFj+2/EuwoEdQKCoKEM2k467WLkYRZuJQx1GFcPL
	OTHER-CDROM-DVDTelemetrySessionCount	CD_ROM Session count
	OTHER-CDROM-DVDTelemetrySessionDuration	CDROM Session Duration
	OTHER-DISK-DVDTelemetrySessionCount	Disk Session count
	OTHER-DISK-DVDTelemetrySessionDuration	Disk Session Duration
	OfficeReliabilityUID	Example: jEfxrd19dUm5nOi/KAJZEA==
	WERUserDisabled	WER Enabled state for a given user
	WMC-CDROM-DVDTelemetrySessionCount	CDROM Session count
	WMC-CDROM-DVDTelemetrySessionDuration	CDROM Session Duration
	WMC-DISK-DVDTelemetrySessionCount	WMC Disk Session count
	WMC-DISK-DVDTelemetrySessionDuration	WMC Disk Session Duration
	WMP-CDROM-DVDTelemetrySessionCount	WMP CDROM Session count
	WMP-CDROM-DVDTelemetrySessionDuration	WMP CDROM Session Duration
	WMP-DISK-DVDTelemetrySessionCount	WMP CDROM Session count
	WMP-DISK-DVDTelemetrySessionDuration	WMP CDROM Session Duration
InventoryMiscellaneousUserAccountTypeEnumeration		Lists the details about the various MSA attached to the system (no user-identifiable information).
	Key	AccountType_{StandardUserHash}
	AccountType	Describes whether the user is a: Local, Domain, MSA or AAD user
	Flags	If bit 0x0002 is set on flags, the account is disabled.
	IsBuiltIn	If IsBuiltIn is true, the account is a built in user.
	Privileges	Privileges defines whether the account is an admin (2), regular user (1) or guest (0).
	StandardUserHash	Standardized user hash from the SID of the user.
InventoryMiscellaneousVolumeInfo		Lists information about a volume, such as drive letter, type, file system, size, etc.
	Key	"VolumeInfo" + {DriveLetter minus :} (or if no drive letter, 2-digit running counter)
	Capacity	Size of the volume in bytes
	DriveLetter	Drive letter assigned to a volume, or empty for volumes without drive letters
	DriveType	What type of drive is this volume?
	EncryptionState	Uses FullVolumeEncryption to get encryption state. Possible options are

	FileSystem	File system on the logical disk
	FreeSpace	Space, in bytes, available on the logical disk
	IsBootVolume	Is this the boot volume?
	IsSystemVolume	Is this the system volume?
InventoryMiscellaneousWinSATMetrics		Lists information for WinSAT which is a key performance indicator.
	Key	Example: WinSATMetrics.000
	MetricCategory	Example: CPUMetrics
	MetricDXVersion	Example: 9
	MetricExpectedFrameCount	Example: 300
	MetricFactor	Example: 0.0
	MetricHeight	Example: 1080
	MetricIOSize	Example: 65536
	MetricKind	Example: Sequential Read
	MetricReason	Example: PASSED
	MetricScore	Example: 4.2
	MetricSubcategory	Example: CompressionMetric
	MetricUnits	Example: MB/s
	MetricValue	Example: 224.35635
	MetricWidth	Example: 1920
InventoryMiscellaneousWAMAccounts		Lists information about Well Known token broker accounts used to measure the MSA attached rate.
	Key	WAM Account
	StandardUserHash	Example: 1471681315
	UserType	User Type for the MSA
	AccountID	AccountID of MSA (hashed)
	DisplayName	The display name of the well-known broker account provider, such Work or school account or Microsoft Account.
InventoryLanguagePackAdd		Represents the count of language packs installed on the system.
	objectInstanceId	Will always be "LanguagePack"
	HasLanguagePack	Is LanguagePackCount >=2? Indicates the presence of a language pack. Example: FALSE
	LanguagePackCount	The count of language packs. Calculated by calling EnumUILanguages with MUI_ALL_INSTALLED_LANGUAGES. Example: 1
InventoryMediaCenterAdd		Represents the decision points (true/false) used to understand if Windows Media Center is used on the system.
	objectInstanceId	Will always be "MediaCenter"
	EverLaunched	Has Media Center ever been launched? Example: FALSE
	HasConfiguredTv	Has the user configured a TV tuner through Media Center? Example: FALSE
	HasExtendedUserAccounts	Are any Media Center Extender user accounts set up? Example: FALSE
	HasWatchedFolders	Are any folders configured for Media Center to watch? Example: FALSE
	IsDefaultLauncher	Is Media Center the default app for opening music or video files? Example: FALSE
	IsPaid	Is the user running a Media Center SKU that implies they paid for Media Center? Example: FALSE
	IsSupported	Does the running OS even support MediaCenter? Example: FALSE
DecisionMediaCenterAdd		Represents the result of all compatibility decisions (true/false) about the presence of Windows Media Center.
	objectInstanceId	Will always be "MediaCenter"
	BlockingApplication	Indicates if there are any application issues that interfere with upgrade due to MediaCenter. Will equal MediaCenterInUse. Example: FALSE
	MediaCenterActivelyUsed	Indicates if MediaCenter is supported on the SKU, has been run at least once, and has the MediaCenterIndicators as true. Example: FALSE
	MediaCenterInUse	Indicates if MediaCenter being used should be bubbled up. This is the top-level indicator and is equal to MediaCenterPaidOrActivelyUsed. Example: FALSE
	MediaCenterIndicators	True if any of the signals indicating media center use are true (default launcher, watched folders, extender accounts, tv tuner). Example: FALSE
	MediaCenterPaidOrActivelyUsed	Equals IsPaid (running on a media center SKU) OR MediaCenterActivelyUsed. Example: FALSE

	NeedsDismissAction	Indicates if there are any dismissible actions coming from MediaCenter. Will equal MediaCenterInUse. Example: FALSE
Gating Events		Gating, or quick-blocking, is used to block upgrades when an application, device, or driver has been detected that could prevent a successful upgrade. Appraiser scans the device against a list of known items that could prevent a successful upgrade. Appraiser scans the device to compare the device to determine if it has the latest quick-block list, updates the list if a new one is available, and then runs the quick-block instructions. These events track the process of checking, running, and successful completion of the quick-blocking run.
GatedDownloadSuccess		Event that indicates that the download of a set of instructions for quick-blocking has succeeded. The download has run a check to see if there are any issues on this device that would make the device ineligible for upgrade. At a future date the scan could be successful if the blocking issue has been fixed.
	Time	The client time of the event.
	CensusId	An ID for the system calculated from the Sqm ID, WER ID, and RAC IDs. Example: {654e6838-9772-4f41-aafc-01642afb4081}.
	Target	The URL of the CAB that was downloaded to run the scan.
GatedNewTarget		Event that indicates that a new set of instructions for quick-blocking is available.
	Time	The client time of the event.
	CensusId	An ID for the system calculated from the Sqm ID, WER ID, and RAC IDs. Example: {654e6838-9772-4f41-aafc-01642afb4081}.
	Target	The URL of the CAB that will be downloaded to run the scan.
GatedRegChange		Event that relays the results of running a set of quick-blocking instructions.
	Time	The client time of the event.
	CensusId	An ID for the system calculated from the Sqm ID, WER ID, and RAC IDs. Example: {654e6838-9772-4f41-aafc-01642afb4081}.
	RegKey	The registry key name for which a result is being sent.
	RegValue	The registry value for which a result is being sent.
	OldData	The previous data in the registry value before the scan ran.
	NewData	The data in the registry value after the scan completed.
GatedRunEnd		Event that indicates that a quick-blocking run has completed and provides the result.
	Time	The client time of the event.
	CensusId	An ID for the system calculated from the Sqm ID, WER ID, and RAC IDs. Example: {654e6838-9772-4f41-aafc-01642afb4081}.
	Target	The URL that provided the CAB that will be used to run the scan.
	Success	Whether or not the scan was successful. Example: 1.
	Result	The hresult from the scan. Example: 0.
GatedRunStart		Event that indicates that a quick-blocking run has started.
	Time	The client time of the event.
	CensusId	An ID for the system calculated from the Sqm ID, WER ID, and RAC IDs. Example: {654e6838-9772-4f41-aafc-01642afb4081}.
	Target	The URL that provided the CAB that will be used to run the scan.
Basic Device Information Events		Basic Device Information Events provide a high level census about the apps, hardware, batteries, display, and other attributes of the device.
Census.App		Retrieves which version of Apps are running on this device.
	CensusVersion	The version of Census that generated the current data for this device.
	IE Version	Retrieves which version of Internet Explorer is running on this device.
Census.Battery		Retrieves information about the type of battery and capacity; and helps to keep a running tally of how many connected standby devices are in use.
	InternalBatteryCapablities	Represents information about what the battery is capable of doing.
	InternalBatteryCapacityCurrent	Represents the battery's current fully charged capacity in mWh (or relative). Compare this value to DesignedCapacity to estimate the battery's wear.
	InternalBatteryCapacityDesign	Represents the theoretical capacity of the battery when new, in mWh.
	IsAlwaysOnAlwaysConnectedCapable	Represents whether the battery enables the device to be AlwaysOnAlwaysConnected . Boolean value.
Census.Camera		Retrieves information about the resolution of camera’s on the device

	FrontFacingCameraResolution	Represents the resolution of the front facing camera in megapixels. If a front facing camera does not exist, then the value is 0.
	RearFacingCameraResolution	Represents the resolution of the rear facing camera in megapixels. If a rear facing camera does not exist, then the value is 0.
Census.Enterprise		Retrieves information about Azure presence, type, and cloud domain use. This information provides an understanding of the use and integration of devices in an enterprise, cloud, and server environment
	AzureOSIDPresent	Represents the field used to identify an Azure machine.
	AzureVMType	Represents whether the instance is Azure VM PAAS, Azure VM IAAS or any other VMs.
	CommercialId	Represents the GUID for the commercial entity which the device is a member of. Will be used to reflect insights back to customers.
	HashedDomain	The hashed representation of the user domain used for login.
	IsCloudDomainJoined	Is this device joined to an Azure Active Directory (AAD) tenant? true/false
	IsDERequirementMet	Represents if the device can do device encryption.
	IsDeviceProtected	Represents if Device protected by BitLocker/Device Encryption
	IsDomainJoined	Indicates whether a machine is joined to a domain.
	IsEDPEnabled	Represents if Enterprise data protected on the device.
	IsMDMEnrolled	Whether the device has been MDM Enrolled or not.
	MPNId	Returns the Partner ID/MPN ID from Regkey. HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\DeployID
	SCCMClientId	This ID correlate systems that send data to Compat Analytics (OMS) and other OMS based systems with systems in an Enterprise SCCM environment.
	ServerFeatures	Represents the features installed on a Windows Server. This can be used by developers and administrators who need to automate the process of determining the features installed on a set of server computers.
	SystemCenterID	The SCCM ID is an anonymized one-way hash of the Active Directory Organization identifier.
	CDJType	Represents the type of cloud domain joined for the machine.
Census.Firmware		BIOS information. Describes the BIOS and startup embedded in the device.
	FirmwareManufacturer	Represents the manufacturer of the device's firmware (BIOS).
	FirmwareReleaseDate	Represents the date the current firmware was released.
	FirmwareType	Represents the firmware type. The various types can be unknown, BIOS, UEFI.
	FirmwareVersion	Represents the version of the current firmware.
Census.Flighting		Retrieves flighting (pre-release build) information. Provides information about devices that are part of Microsoft flighting releases – limited releases with incremental changes or improvements to customers participating in improvement testing and feedback programs.
	DeviceSampleRate	Represents the telemetry sample rate assigned to the machine
	EnablePreviewBuilds	Represents the field is used to enable flighting (pre-release build) of preview builds on a machine.
	FlightIds	Represents the list of the different flights (pre-release builds) on this device.
	FlightingBranchName	Represents the name of the branch that the device is flighting (pre-release builds).
	IsFlightsDisabled	Represents if the device is participating in flighting (pre-release builds).
	SSRK	Retieves the mobile targetting settings.
	MSA_Accounts	Represents a list of hashed IDs of the Microsoft Accounts that are flighting (pre-release builds) on this device.
Census.Hardware		Retrieves information about the device, including hardware type; OEM brand, model line, model; telemetry level setting; and TPM support.
	SoCName	Firmware manufacturer of phone.
	DeviceForm	This indicates the form as per the Device classification.
	DigitizerSupport	Friendly name of Digitizer Support. If deviceFamily is of Windows Mobile type, we default digitizer support to be true.
	ChassisType	Retrieves a numeric representation of what type of chassis the machine has. A value of 0 means xx, 1 means yy, 2 means zz .
	ComputerHardwareID	Identifies a class of machines, as a hash of 9 different SMBIOS fields.
	DeviceColor	Indicates the color of the LUMIA phones
	VoiceSupported	Indicates if the device has a cell radio capable of making voice calls (in simple terms - a phone).

	PowerPlatformRole	Indicates the OEM preferred power management profile. This value helps identify the basic form factor of the device.
	TPMVersion	Retrieves the supported Trusted Platform Module (TPM), 0 if no TPM is present.
	OEMManufacturerName	Manufacturer Name of machine. The OEMName for an inactive machine is not reprocessed even if the clean OEM name is changed at a later date.
	OEMModelNumber	Model number of machine.
	OEMModelName	Model name of machine.
	OEMModelSKU	SKU of machine as defined by manufacturer.
	OEMOptionalIdentifier	Microsoft assigned value representing a specific OEM subsidiary.
	OEMSerialNumber	Serial number of machine as set by manufacturer. Sourced from SMBIOS.
	PhoneManufacturer	Friendly name of the phone OEM. This is useful when the ODM (Original Device Manufacturer such as Foxconn,LongCheer) is not the same as the OEM.
	DeviceName	Device Name of the machine as set by the user.
	OEMDigitalMarkerFileName	Retrieves the name of the file placed in the \Windows\system32\drivers directory that specifies the OEM and model name of the machine.
	DUID	Device Unique ID. This is the Microsoft issued unique IDs of phones.
	InventoryId	Device ID used for Compatibility testing.
	StudyID	Used to identify retail/non retail phones.
	TelemetryLevel	The telemetry level the user has opted into, such as Basic or Enhanced.
	TelemetrySettingAuthority	Determines who set the telemetry level (GPM, MDM, User)
	OEMModelBaseBoard	Indicates the baseboard model used by the OEM's
	OEMModelSystemFamily	Indicates the System family set on the machine by OEM's.
Census.VM		Retrieves information about if virtualization is enabled on machine and it's various characteristics.
	IOMMUPresent	Represents if an input/output memory management unit (IOMMU) is present.
	SLATSupported	Represents whether Second Level Address Translation (SLAT) is supported by the hardware.
	VirtualizationFirmwareEnabled	Represents whether virtualization is enabled in the firmware.
	HyperVisor	Retrieves whether the current OS is running on top of a Hypervisor.
	IsVirtualDevice	Retrieves that when the Hypervisor is Microsoft's Hyper-V Hypervisor or other Hv#1 Hypervisor, this field will be set to FALSE for the Hyper-V host OS and TRUE for any guest OS's. This field should not be relied upon for non-Hv#1 Hypervisors.
Census.Memory		Retrieves information about the memory attributes of the device, including ROM and RAM.
	TotalPhysicalRAM	Represents the physical RAM in MB.
	TotalVisibleMemory	Represents the visible memory -memory that is not reserved by the system.
Census.Network		Mobile and cellular network information. Retrieves information about the mobile service provider, network, and device ID, as well as service cost factors.
	IMEI0IMEI1	Represents the International Mobile Station Equipment Identity. This number is usually unique and used by the mobile operator to distinguish different phone hardware. Microsoft does not have access to mobile operator billing data so collecting this data does not expose or identify the user. The two fields represent phone with dual sim coverage.
	MobileOperatorBilling	Represents the telephone company that provides services for mobile phone users.
	MobileOperatorCommercialized	Represents which reseller and geography the phone is commercialized for. This is the set of values on the phone for who and where it was intended to be used. For example, the commercialized mobile operator code AT&T in the US would be ATT-US.
	MobileOperatorNetwork0MobileOperatorNetwork1	Represents the operator of the current mobile network that the device is used on. (AT&T, T-Mobile, Vodafone). The two fields represent phone with dual sim coverage.
	NetworkCost	Represents the network cost associated with a connection.
	MCC0MCC1	Represents the Mobile Country Code (MCC). It used with the Mobile Network Code (MNC) to uniquely identify a mobile network operator. The two fields represent phone with dual sim coverage.
	MEID	Represents the Mobile Equipment Identity (MEID). MEID is a worldwide unique phone ID assigned to CDMA phones. MEID replaces electronic serial number (ESN), and is equivalent to IMEI for GSM and WCDMA phones. Microsoft does not have access to mobile operator billing data so collecting this data does not expose or identify the user.

	MNC0MNC1	Retrieves the Mobile Network Code (MNC). It used with the Mobile Country Code (MCC) to uniquely identify a mobile network operator. The two fields represent phone with dual sim coverage.
	SPN0SPN1	Retrieves the Service Provider Name (SPN). For example, these might be AT&T, Sprint, T-Mobile, or Verizon. The two fields represent phone with dual sim coverage.
Census.WU		Retrieves the information about the windows update server and other Appstore policies.
	AppStoreAutoUpdate	Retrieves the Appstore settings for auto upgrade. (Enable/Disabled).
	AppStoreAutoUpdateMDM	Retrieves the App Auto Update value for MDM 0 - Disallowed 1 - Allowed 2 - Not configured Default: [2] "Not configured"
	AppStoreAutoUpdatePolicy	Retrieves the Windows Store App Auto Update group policy setting
	DelayUpgrade	Retrieves the Windows upgrade flag for delaying upgrades.
	OSWUAutoUpdateOptions	Retrieves the auto update settings on the device.
	UpdateServiceURLConfigured	Retrieves if the device is managed by Windows Server Update Services (WSUS).
	WUDeferUpdatePeriod	Retrieves if deferral is set for Updates
	WUDeferUpgradePeriod	Retrieves if deferral is set for Upgrades
	WUDODownloadMode	Retrieves whether DO is turned on and how to acquire/distribute updates Delivery Optimization (DO) allows users to deploy previously downloaded WU updates to other devices on the same network.
	WUMachineId	Retrieves the Windows Update (WU) Machine Identifier.
	WUPauseState	Retrieves WU setting to determine if updates are paused
	WUServer	Retrieves the HTTP(S) URL of the WSUS server that is used by Automatic Updates and API callers (by default).
Census.OS		Retrieves operating system attributes, such as Windows edition and whether it is a virtual device. Retrieves information about the version of Windows currently in use; how and when Windows was originally installed; operating system locale; and update service configuration.
	GenuineState	Retrieves the ID Value specifying the OS Genuine check.
	IsPortableOperatingSystem	Retrieves whether OS is running via "Windows-To-Go".
	IsSecureBootEnabled	Retrieves whether Boot chain is signed under UEFI.
	OSEdition	Retrieves the version of the current OS.
	InstallationType	Retrieves the type of OS installation. (Clean, Upgrade, Reset, Refresh, Update).
	OSInstallDateTime	Retrieves the date the OS was installed using ISO 8601 (Date part) == yyyy-mm-dd
	OSInstallType	Retrieves a numeric description of what install was used on the device i.e. clean, upgrade, refresh, reset, etc
	OSOOBEDateTime	Retrieves Out of Box Experience (OOBE) Date in Coordinated Universal Time (UTC).
	OSSKU	Retrieves the Friendly Name of OS Edition.
	OSTimeZoneBiasInMins	Retrieves the time zone set on machine.
	OSUILocale	Retrieves the locale of the UI that is currently used by the OS.
	RACw7Id	Retrieves the Microsoft Reliability Analysis Component (RAC) Win7 Identifier. RAC is used to monitor and analyze system usage and reliability.
	CompactOS	Indicates if the Compact OS feature from Win10 is enabled.
	Signature	Retrieves if it is a signature machine sold by Microsoft store.
	IsDeviceRetailDemo	Retrieves if the device is running in demo mode.
	OA3xOriginalProductKey	Retrieves the License key stamped by the OEM to the machine.
	ProductKeyID2	Retrieves the License key if the machine is updated with a new license key.
	ServiceMachineIP	Retrieves the IP address of the KMS host used for anti-piracy.
	ActivationChannel	Retrieves the retail license key or Volume license key for a machine.
	LicenseStateReason	Retrieves why (or how) a system is licensed or unlicensed. The HRESULT may indicate an error code that indicates a key blocked error, or it may indicate that we are running an OS License granted by the MS store.
	LanguagePacks	The list of language packages installed on the device.

	InstallLanguage	The first language installed on the user machine.
	ServiceProductKeyID	Retrieves the License key of the KMS
	SharedPCMode	Returns Boolean for education devices used as shared cart
	IsEduData	Returns Boolean if the education data policy is enabled.
	SLICVersion	Returns OS type/version from SLIC table.
	ProductActivationTime	Returns the OS Activation time for tracking piracy issues.
	ProductActivationResult	Returns Boolean if the OS Activation was successful.
	OSSubscriptionTypeId	Returns boolean for enterprise subscription feature for selected PRO machines.
	OSSubscriptionStatus	Represents the existing status for enterprise subscription feature for PRO machines.
	SLICStatus	Whether a SLIC table exists on the device.
Census.Processor		Retrieves information about the processor architecture, speed, number of cores, manufacturer, and model number.
	ProcessorArchitecture	Retrieves the processor architecture of the installed operating system. The complete list of values can be found in DimProcessorArchitecture.
	ProcessorClockSpeed	Retrieves the clock speed of the processor in MHz.
	ProcessorCores	Retrieves the number of cores in the processor.
	ProcessorManufacturer	Retrieves the name of the processor's manufacturer.
	ProcessorModel	Retrieves the name of the processor model.
	SocketCount	Number of physical CPU sockets of the machine.
	ProcessorPhysicalCores	Number of physical cores in the processor.
Census.Storage		Retrieves the total capacity of the system volume and primary disk.
	PrimaryDiskTotalCapacity	Retrieves the amount of disk space on the primary disk of the device in MB.
	PrimaryDiskType	Retrieves an enumerator value of type STORAGE_BUS_TYPE that indicates the type of bus to which the device is connected. This should be used to interpret the raw device properties at the end of this structure (if any).
	SystemVolumeTotalCapacity	Retrieves the size of the partition that the System volume is installed on in MB.
Census.Userdefault		This Census User Default telemetry point collects information on the current user's default preferences for browser and several of the most popular extensions and protocols
	DefaultBrowserProgId	The ProgramId of the current user's default browser
	DefaultApp	The current uer's default program selected for the following extension or protocol: .html,.htm,.jpg,.jpeg,.png,.mp3,.mp4, .mov,.pdf
Census.UserDisplay		Retrieves information about the logical/physical display size, resolution and number of internal/external displays. Additionally there are also details around the VRAM on the system.
	VRAMDedicated	Represents the video RAM in MB.
	VRAMDedicatedSystem	Represents the amount of memory on the dedicated video card.
	VRAMSharedSystem	Represents the amount of RAM memory that the video card can use.
	NumberofExternalDisplays	Retrieves the number of external displays connected to the machine
	InternalPrimaryDisplayLogicalDPIX	Represents the logical DPI in the x-direction of the internal display.
	InternalPrimaryDisplayLogicalDPIY	Represents the logical DPI in the y-direction of the internal display.
	InternalPrimaryDisplayPhysicalDPIX	Represents the physical DPI in the x-direction of the internal display.
	InternalPrimaryDisplayPhysicalDPIY	Represents the physical DPI in the y-direction of the internal display.
	InternalPrimaryDisplayResolutionHorizontal	Represents the number of pixels in the horizontal direction of the internal display.
	InternalPrimaryDisplayResolutionVertical	Represents the number of pixels in the vertical direction of the internal display.
	InternalPrimaryDisplaySizePhysicalH	Represents the physical horizontal length of the display in mm. Used for calculating the diagonal length in inches .
	InternalPrimaryDisplaySizePhysicalY	Represents the physical vertical length of the display in mm. Used for calculating the diagonal length in inches
	InternalPrimaryDisplayType	Represents the type of technology used in the monitor, such as Plasma, LED, LCODS, etc.
	NumberofInternalDisplays	Represents the number of internal displays in a device.
Census.UserNLS		Retrieves the default app language, input and display language preferences set by the user.
	DefaultAppLanguage	The current user Default App Language.
	DisplayLanguage	The current user preferred Windows Display Language.

	HomeLocation	The current user location, populated using GetUserGeoId() function, which returns location at the country or region level.
	KeyboardInputLanguages	The Keyboard input languages installed on the device.
	SpeechInputLanguages	The Speech Input languages installed on the device.
Census.Xbox		Xbox Console information like Serial Number and DeviceId.
	XboxConsolePreferredLanguage	Retrieves the preferred language selected by the user on Xbox console.
	XboxConsoleSerialNumber	Retrieves the serial number of the Xbox console.
	XboxLiveDeviceId	Retrieves the unique device id of the console.
	XboxLiveSandboxId	Retrieves the developer sandbox id if the device is internal to MS.
Setup360telemetry.downlevel		This provider sends events for OS Updates and Upgrades from Windows 7.X, Windows 8.X, and Windows 10. Specifically the Setup360telemetry.downlevel event indicates that a specific device has invoked the downlevel phase of the upgrade.
	ClientId	In the Windows Update scenario, this will be the Windows Update client ID that is passed to Setup. In Media setup, default value is Media360, but can be overwritten by the caller to a unique value (being leveraged by China Partners)
	InstanceId	Unique GUID that identifies each instance of setuphost.exe
	ReportId	In the Windows Update scenario, this is the updateID that is passed to Setup. In media setup, this is the GUID for the install.wim.
	WuId	This is the Windows Update Client ID. In the Windows Update scenario, this is the same as the clientId.
	TestId	String to uniquely identify a group of events, such as all events for the upgrade fair.
	State	Exit state of given Setup360 run (succeeded, failed, blocked, cancelled)
	HostOsSkuName	OS SKU which is running Setup360 instance (downlevel OS)
	HostOsBuildNumber	Build number of downlevel OS
	Setup360Scenario	Setup360 flow type (Boot, Media, Update, MCT)
	Setup360Mode	Phase of Setup360 (Predownload, Install, Finalize, Rollback etc.)
	Setup360Result	Result of Setup360 (HRESULT used to diagnose errors)
	Setup360Extended	Extension of result - more granular information about phase/action when the potential failure happened
	SetupVersionBuildNumber	Build number of Setup360 (build number of target OS).
Setup360telemetry.finalize		This provider sends events for OS Updates and Upgrades from Windows 7.X, Windows 8.X, and Windows 10. Specifically the Setup360Telemetry.finalize event indicates that a specific device has invoked the finalize phase of the upgrade.
	ClientId	In the Windows Update scenario, this will be the Windows Update client ID that is passed to Setup. In Media setup, default value is Media360, but can be overwritten by the caller to a unique value (being leveraged by China Partners)
	InstanceId	Unique GUID that identifies each instance of setuphost.exe
	ReportId	In the Windows Update scenario, this is the updateID that is passed to Setup. In media setup, this is the GUID for the install.wim.
	WuId	This is the Windows Update Client ID. In the Windows Update scenario, this is the same as the clientId.
	TestId	String to uniquely identify a group of events.
	State	Exit state of given Setup360 run (succeeded, failed, blocked, cancelled)
	HostOsSkuName	OS SKU which is running Setup360 instance (downlevel OS)
	HostOsBuildNumber	Build number of downlevel OS
	Setup360Scenario	Setup360 flow type (Boot, Media, Update, MCT)
	Setup360Mode	Phase of Setup360 (Predownload, Install, Finalize, Rollback etc.)
	Setup360Result	Result of Setup360 (HRESULT used to diagnose errors)
	Setup360Extended	Extension of result - more granular information about phase/action when the potential failure happened
	SetupVersionBuildNumber	Build number of Setup360 (build number of target OS).

Setup360telemetry.postrebootinstall		This provider sends events for OS Updates and Upgrades from Windows 7.X, Windows 8.X, and Windows 10. Specifically the Setup360Telemetry.postrebootinstall event indicates that a specific device has invoked the postrebootinstall phase of the upgrade.
	ClientId	In the Windows Update scenario, this will be the Windows Update client ID that is passed to Setup. In Media setup, default value is Media360, but can be overwritten by the caller to a unique value (being leveraged by China Partners)
	InstanceId	Unique GUID that identifies each instance of setuphost.exe
	ReportId	In the Windows Update scenario, this is the updateID that is passed to Setup. In media setup, this is the GUID for the install.wim.
	WuId	This is the Windows Update Client ID. In the Windows Update scenario, this is the same as the clientId.
	TestId	String to uniquely identify a group of events, such as all events for the upgrade fair.
	State	Exit state of given Setup360 run (succeeded, failed, blocked, cancelled)
	HostOsSkuName	OS SKU which is running Setup360 instance (downlevel OS)
	HostOsBuildNumber	Build number of downlevel OS
	Setup360Scenario	Setup360 flow type (Boot, Media, Update, MCT)
	Setup360Mode	Phase of Setup360 (Predownload, Install, Finalize, Rollback etc.)
	Setup360Result	Result of Setup360 (HRESULT used to diagnose errors)
	Setup360Extended	Extension of result - more granular information about phase/action when the potential failure happened
	SetupVersionBuildNumber	Build number of Setup360 (build number of target OS).
Setup360telemetry.predownloadquiet		This provider sends events for OS Updates and Upgrades from Windows 7.X, Windows 8.X, and Windows 10. Specifically the Setup360Telemetry.PredownloadQuiet event indicates that a specific device has invoked the predownload quiet phase of the upgrade.
	ClientId	In the Windows Update scenario, this will be the Windows Update client ID that is passed to Setup. In Media setup, default value is Media360, but can be overwritten by the caller to a unique value (being leveraged by China Partners)
	InstanceId	Unique GUID that identifies each instance of setuphost.exe
	ReportId	In the Windows Update scenario, this is the updateID that is passed to Setup. In media setup, this is the GUID for the install.wim.
	WuId	This is the Windows Update Client ID. In the Windows Update scenario, this is the same as the clientId.
	TestId	String to uniquely identify a group of events, such as all events for the upgrade fair.
	State	Exit state of given Setup360 run (succeeded, failed, blocked, cancelled)
	HostOsSkuName	OS SKU which is running Setup360 instance (downlevel OS)
	HostOsBuildNumber	Build number of downlevel OS
	Setup360Scenario	Setup360 flow type (Boot, Media, Update, MCT)
	Setup360Mode	Phase of Setup360 (Predownload, Install, Finalize, Rollback etc.)
	Setup360Result	Result of Setup360 (HRESULT used to diagnose errors)
	Setup360Extended	Extension of result - more granular information about phase/action when the potential failure happened
	SetupVersionBuildNumber	Build number of Setup360 (build number of target OS).
Setup360telemetry.preinstallquiet		This provider sends events for OS Updates and Upgrades from Windows 7.X, Windows 8.X, and Windows 10. Specifically the Setup360Telemetry.preinstallQuiet event indicates that a specific device has invoked the preinstall quiet phase of the upgrade.
	ClientId	In the Windows Update scenario, this will be the Windows Update client ID that is passed to Setup. In Media setup, default value is Media360, but can be overwritten by the caller to a unique value (being leveraged by China Partners)
	InstanceId	Unique GUID that identifies each instance of setuphost.exe
	ReportId	In the Windows Update scenario, this is the updateID that is passed to Setup. In media setup, this is the GUID for the install.wim.
	WuId	This is the Windows Update Client ID. In the Windows Update scenario, this is the same as the clientId.

	TestId	String to uniquely identify a group of events.
	State	Exit state of given Setup360 run (succeeded, failed, blocked, cancelled)
	HostOsSkuName	OS SKU which is running Setup360 instance (downlevel OS)
	HostOsBuildNumber	Build number of downlevel OS
	Setup360Scenario	Setup360 flow type (Boot, Media, Update, MCT)
	Setup360Mode	Phase of Setup360 (Predownload, Install, Finalize, Rollback etc.)
	Setup360Result	Result of Setup360 (HRESULT used to diagnose errors)
	Setup360Extended	Extension of result - more granular information about phase/action when the potential failure happened
	SetupVersionBuildNumber	Build number of Setup360 (build number of target OS).
Setup360telemetry.unexpectedevent		This provider sends events for OS Updates and Upgrades from Windows 7.X, Windows 8.X, and Windows 10. Specifically the Setup360Telemetry.uexpected event indicates that a specific device has invoked the unexpected event phase of the upgrade.
	ClientId	In the Windows Update scenario, this will be the Windows Update client ID that is passed to Setup. In Media setup, default value is Media360, but can be overwritten by the caller to a unique value (being leveraged by China Partners)
	InstanceId	Unique GUID that identifies each instance of setuphost.exe
	ReportId	In the Windows Update scenario, this is the updateID that is passed to Setup. In media setup, this is the GUID for the install.wim.
	WuId	This is the Windows Update Client ID. In the Windows Update scenario, this is the same as the clientId.
	TestId	String to uniquely identify a group of events.
	State	Exit state of given Setup360 run (succeeded, failed, blocked, cancelled)
	HostOsSkuName	OS SKU which is running Setup360 instance (downlevel OS)
	HostOsBuildNumber	Build number of downlevel OS
	Setup360Scenario	Setup360 flow type (Boot, Media, Update, MCT)
	Setup360Mode	Phase of Setup360 (Predownload, Install, Finalize, Rollback etc.)
	Setup360Result	Result of Setup360 (HRESULT used to diagnose errors)
	Setup360Extended	Extension of result - more granular information about phase/action when the potential failure happened
	SetupVersionBuildNumber	Build number of Setup360 (build number of target OS).
Setup360Telemetry.Setup360		Retrieves events for OS deployment scenarios
	ClientId	Retrieves the upgrade ID: Upgrades via WU - specifies the WU clientID Upgrades via China Partners - specifies a unique string that classifies these events All other deployment - static string.
	FieldName	Retrieves the event name/data point
	FlightData	Specifies a unique identifier for each group of flights (pre-release builds)
	InstanceId	Retrieves a unique identifier for each instance of a setup session
	ReportId	Retrieves the report ID:
	ScenarioId	Retrieves the deployment scenario
	Value	Retrieves the value associated with the corresponding FieldName
SetupPlatformTelEvent		This service retrieves events generated by SetupPlatform, the engine that drives the various deployment scenarios.
	FieldName	Retrieves the event name/data point. Examples: InstallStartTime, InstallEndtime, OverallResult etc .
	GroupName	Retrieves the groupname the event belongs to. Example: Install Information, DU Information, Disk Space Information etc.
	Value	Retrieves the value associated with the corresponding event name (Field Name). For example: For time related events this will include the system time.
	Name	Retrieves the activity name; for Dynamic Update operations events are categorized based on an activity name.
SetupPlatformTelActivityEvent		This event is used to generate a unique ID that can be used to bind Setup Platform events together
	ActivityId	Provides a unique Id to correlate events that occur between a activity start event, and a stop event

	ActivityName	Provides a friendly name of the package type that belongs to the ActivityId (Setup, LanguagePack, GDR, Driver, etc.)
TelClientSynthetic.ConnectivityHeartBeat_0		This event is used to determine the connectivity status of the Connected User Experience and Telemetry component that uploads telemetry events. If an unrestricted free network (for example Wi-Fi) is available, this event updates the last successful upload time. Else, it checks if a Connectivity Heartbeat event was fired in the past 24 hours, and if not, it fires an event. A Connectivity Heartbeat event also fired when a device recovers from costed network to free network.
	LastConntectivityLossTime	Retrieves the last time the device lost free network.
	NetworkState	Retrieves the network state: 0 = No network. 1 = Restricted network. 2 = Free network.
	NoNetworkTime	Retrieves the time spent with no network (since the last time) in seconds.
	RestrictedNetworkTime	Retrieves the time spent on a metered (cost restricted) network in seconds.
TelClientSynthetic. HeartBeat_5		This event contains statistics about the health and quality of the telemetry data from the given device. Also enables data analysts to determine how "trusted" the data is from a given device. Fires every 30 minutes and linked to the previous heartbeat event using the PreviousHeartBeatTime parameter.
	ConsumerDroppedCount	Retrieves the number of events dropped in the consumer buffer.
	CriticalOverflowEntersCounter	Retrieves the number of times we have entered critical overflow mode.
	DbCriticalDroppedCount	Retrieves the number of events marked with critical persistence that have been dropped due to disk size and/or bandwidth limitations.
	DbDroppedCount	Retrieves the number of events dropped by the local service due to disk size and/or bandwidth limitations.
	DecodingDroppedCount	Retrieves the number of events dropped due to ETW Decoding failures.
	EnteringCriticalOverflowDroppedCounter	Retrieves the sum of bytes dropped from normal buffers due to entering critical overflow.
	EtwDroppedCount	Retrieves the number of events dropped in ETW buffer.
	EventSubStoreResetCounter	Retrieves the number of times an event substore was reset.
	EventSubStoreResetSizeSum	Retrieves the sum of sizes (in bytes) of all event substores reset.
	EventsUploaded	Retrieves the total number of events uploaded since last successful upload.
	InvalidHttpCodeCount	Retrieves the total number of failed responses from the Microsoft Data Management Service servers.
	LastEventSizeOffender	Retrieves the name of the last event to exceed the allowed single-upload event payload size.
	LastInvalidHttpCode	Retrieves the last HTTP error received from Microsoft Data Management Service servers when attempting to upload.
	MaxInUseScenarioCounter	Retrieves the maximum # of concurrent in use scenarios during this timeframe since last heartbeat.
	PreviousHeartBeatTime	Retrieves the timestamp of the last heart beat which was fired.
	SettingsHttpAttempts	Retrieves the number of times the Connected User Experience and Telemetry component attempted to connect to OneSettings.
	SettingsHttpFailures	Retrieves the number of times the Connected User Experience and Telemetry component failed to connect to OneSettings.
	ThrottledDroppedCount	Retrieves the number of events which were dropped due to throttling.
	UploaderDroppedCount	Retrieves the number of dropped events due to the upload buffer.
	VortexHttpAttempts	Retrieves the number of attempts the Connected User Experience and Telemetry component attempted to upload data to Microsoft Data Management servers.
	VortexHttpeFailures4xx	Retrieves the count, which increases each time the client receives a 400-499 code back from the service.
TelClientSynthetic. AbnormalShutdown_0		This event is logged to identify boot ids for which a normal clean shutdown was not observed.
	AbnormalShutdownBootId	Retrieves the Boot ID for which the abnormal shutdown was observed.
	CurrentBootId	Retrieves the current boot ID.
	LastSuccessfullyShutdownBootId	Retrieves the last successfully/cleanly shutdown boot ID.
Microsoft Emulator		The Microsoft Emulator starts up a HyperV Virtual Machine (VM). Many of the events are related to making sure the VM booted properly.
	BootCompleted	Represents how long it took the emulated hardware to boot.
	ButtonClicked	Represents the User Interface interacted with.
	ConnectedToGuesd	Represents the connection to emulated hardware.
	ExceptionThrown	Represents the internal error state hit.

	GuestNotificationsConnectedTo	Represents the connection to emulated hardware sensors.
	GuestResolutionIndicated	Represents the emulated hardware resolution.
	HotkeyPressed	Represents the User Interface interacted with.
	InputModeInitialized	Represents the Mouse, Touch, or Multi-Touch initialized.
	InputUsed	Represents the Mouse, Touch, or Keyboard Input type selected.
	MicrophoneServiceConnectedTo	Represents the connection to emulated hardware fake microphone.
	RemoteFxAvailability	Represents the emulated hardware graphics acceleration supported.
	RemotedFxForEveryXde	Represents the emulated hardware graphics acceleration enabled or disabled.
	SensorsEnabled	Represents the emulated hardware sensors enabled.
	SkinLoaded	Represents the User Interface Device Image Loaded.
	SystemReady	Represents how long it took the emulated hardware to be ready.
	TabClicked	Represents the User Interface interacted with.
	UiShown	How long it took the User Interface to appear.
	VirtualMachineStateChanged	Represents the Virtual Machine changed state.
	VmStarted	Represents the Virtual Machine started.
	XdeStarted	Represents the emulator started with various options.
	XdeStopped	Represents the emulator stopped with exit code.
Microsoft.Windows.Kits.Bootstrapper		These events provide information about kit installation (standalone SDK, ADK, HLk or WDK) on a downlevel machine to see if the kit installed successfully, and what options were selected.
Microsoft.Windows.Kits.Bootstrapper.SetupFinished		This event retrieves information about the completion of setup of the installation package, and about errors that may have occurred during setup.
	commandLineError	Retrieves error in kitsetup command line.
	exitCode	Retrieves exit code from kit installer.
	exitCodeString	Retrieves string explanation of exit code
	engineError	
	engineErrorMessage	Retrieves error message upon completion of kit install.
	engineErrorStage	Retrieves the install stage the error occurred if applicable.
	pathResolutionFailed	Retrieves the Boolean value indicating if the media for installation was present.
	associatedUserOption	Retrieves which feature was being installed when the error occurred.
	associatedPayloadId	
	associatedPackageId	
	associatedPackageError	
	associatedPackageErrorString	
	vsInstalled	
Microsoft.Windows.Kits.Bootstrapper.BurnErrorReported		This event retrieves information about installation failures.
	errorCode	Retrieves the error code of the installation failure, as a hex number.
	errorType	Retrieves the type of installation error, represented by a number.
	packageId	Retrieves the name of the installation package that failed.
	errorMessage	Retrieves the error message displayed to the user when installation fails
Microsoft.Windows.Kits.Bootstrapper.ExceptionHit		This event retrieves information about when an exception is hit during installation.
	Hash	Retrieves the hash code number of the exception.
	type	Retrieves the type of exception, for example “Win32Exception.”
	message	Retrieves the message displayed for this exception, for example "Failed calling HttpSendRequest()".
	stack	Retrieves information about where the exception occurred in the installation process.
	burnResult	Retrieves the result of the installation attempt. For example, “None,” if installation was not able to complete.
Microsoft.Windows.Kits.Bootstrapper.SetupStarted		Retrieves information about the installation kit being installed such as the name, version, and type.
	kitName	Retrieves the name of kit bundle being installed: e.g. BundleID.ADK, Bundle.WDK, BundleID.SDK
	kitVersion	Retrieves the version number of the Windows kit being installed.
	bootstrapperVersion	Retrieves the version number stamped inside of the kitsetup.exe.
	workflow	Retrieves the installation action being taken: Install, Modify, Repair, Uninstall.
	runningFromDownloadedLocation	Retrieves the Boolean value indicating if running from website or locally.
	commandLineBitmap	Retrieves the bitmap of command line parameters passed to the exe.

	selectedOptions	Retrieves the mask of kit options selected for installation or removal.
Microsoft.Windows.WorkFolders.ChangeEnumeration		Retrieves information about kit installation (standalone SDK, ADK, HLk or WDK) on a downlevel machine to see if the kit installed successfully, and what options were selected.
	DirectoryCount	Retrieves a count of directories in Work Folders path. Recursive
	TaskDuration	Retrieves time taken by “Change Enumeration” task on the machine, in milliseconds.
	FileCount	Retrieves a count of files in Work Folders path, recursive.
	HResult	Retrieves the exit code returned for “Change Enumeration” task.
	DataSize	Retrieves total data size (bytes) in Work Folders directory.
	MaxFileSize	Retrieves the maximum file size (bytes) in the Work Folders directory.
	MinFileSize	Retrieves the minimum file size (bytes) in the Work Folders directory.
	Upto1KB	Retrieves the count for file sizes < 1KB.
	Upto10KB	Retrieves the count for file sizes < 10KB.
	Upto100KB	Retrieves the count for file sizes < 100KB.
	Upto1MB	Retrieves the count for file sizes < 1MB.
	Upto10MB	Retrieves the count for file sizes < 10MB.
	Upto100MB	Retrieves the count for file sizes < 100MB.
	Upto1GB	Retrieves the count for file sizes < 1GB.
	Upto10GB	Retrieves the count for file sizes < 10GB.
	10GBPlus	Retrieves the count for file sizes >= 10GB.
Microsoft.Windows.WorkFolders.ChangeTracker		This telemetry event captures the exit code for the change detection task.
	HResult	Retrieves the exit code returned for the task.
Microsoft.Windows.WorkFolders.LocalProvider		This telemetry event captures the task name and exit code for a sync session on local.
	HResult	Retrieves the exit code returned for the task.
	CommandName	Retrieves the task name.
Microsoft.Windows.WorkFolders.RemoteProvider		This telemetry event captures the task name and exit code for a sync session on remote.
	HResult	Retrieves the exit code returned for the task.
	CommandName	Retrieves the task name.
Microsoft.Windows.WorkFolders.PerItemError		This telemetry event helps Microsoft understand the issues encountered specific to a file during upload/download.
	SyncDirection	Retrieves whether the file was uploaded or downloaded.
	HResult	Retrieves the exit code returned for the task.
Microsoft.Windows.WorkFolders.SyncSession		This telemetry event helps Microsoft understand the session specific issues encountered.
	SyncHint	Retrieves how the sync was triggered. (Upload/Download/Reconciliation/Forced).
	SyncReconcilePathHit	Retrieves if reconciliation is needed (True/False).
	SyncReconileSkipped	Retrieves if Reconciliation task is skipped (True/False).
	EncryptionPolicy	Retrieves the encryption policy status (On/Off).
	ReconcileTaskDuration	Retrieves the time taken for the reconciliation task (in Milliseconds).
	Task Duration	Retrieves the total time taken for the sync task (in Milliseconds).
	HResult	Retrieves the exit code returned for the sync task.
Microsoft.Windows.WorkFolders.SyncStatistics		This telemetry event captures sync session statistics to help understand the usage/perf counters.
	SessionDuration	Retrieves the time taken to complete the sync session (in seconds).
	SessionType	Retrieves the type of the session – Regular or Full Enumeration.
	SyncDirection	Retrieves whether it was an upload or download session.
	FilesTransferred	Retrieves the count of files in the session.
	BatchCount	Retrieves the count of batch in the session.
Microsoft.Windows.App.Browser		**These events provide data on a user’s Internet Explorer activity.**
Microsoft.Windows.App.Browser.IECrashInternet (Internet + Restricted Zone)		These events provide data on a user’s Internet Explorer activity.
	url	URL of the browsed site where a crash was encountered. Query string is removed from the URL.
Microsoft.Windows.App.Browser.IECrashIntranet (Intranet +Trusted + Local File Zone)		These events provide data on a user’s Internet Explorer activity.
	url	URL of the browsed site where a crash was encountered. Query string is removed from the URL.
Microsoft.Windows.App.Browser.IEHangInternet (Internet + Restricted Zone)		These events provide data on a user’s Internet Explorer activity.
	url	URL of the browsed site where a browser hang was encountered. Query string is removed from the URL.
Microsoft.Windows.App.Browser.IEHangIntranet (Intranet + Trsuted + Local File Zone)		These events provide data on a user’s Internet Explorer activity.

url	URL of the browsed site where a browser hang was encountered. Query string is removed from the URL.
Microsoft.Windows.App.Browser.IENavFailureInternet (Internet + Restricted Zone)	These events provide data on a user's Internet Explorer activity.
url	URL of the browsed site where a navigation failure occured. Query string is removed from the URL.
Microsoft.Windows.App.Browser.IENavFailureIntranet (Intranet + Trsuted + Local File Zone)	These events provide data on a user's Internet Explorer activity.
url	URL of the browsed site where a navigation failure occured. Query string is removed from the URL.
Microsoft.Web.Platform	**These events provide data on a user's Internet Explorer activity.**
Microsoft.Web.Platform.OMSMarkupDataInternet (Internet + Restricted Zone)	These events provide data on a user's Internet Explorer activity.
url	URL of the browsed site. Query string is removed from URL.
domain	Domain of the URL of the browsed site.
docMode	Document mode used by Internet Explorer for this page.
docModeReason	Describes why a document mode was set by Internet Explorer for this page.
layoutMode	Layout mode of the page.
browserStateReason	Describes the reasons this page loaded in a given layout mode.
emieNavigation	Indicates if this page was loaded in Enterprise Mode.
secZone	Security zone of the browsed site.
Microsoft.Web.Platform.OMSMarkupDataIntranet (Intranet + Trusted + Local File Zone)	These events provide data on a user's Internet Explorer activity.
url	URL of the browsed site. Query string is removed from URL.
domain	Domain of the URL of the browsed site.
docMode	Document mode used by Internet Explorer for this page.
docModeReason	Describes why a document mode was set by Internet Explorer for this page.
layoutMode	Layout mode of the page.
browserStateReason	Describes the reasons this page loaded in a given layout mode.
emieNavigation	Indicates if this page was loaded in Enterprise Mode.
secZone	Security zone of the browsed site.
Microsoft.Web.Platform.VersionManagementInternet (Internet + Restricted Zone)	These events provide data on a user's Internet Explorer activity.
AbsoluteUri	URL of the site that loaded this ActiveX control. Query string is removed from URL.
Entry	The GUID of the control being logged.
FileName	Name of the file being loaded.
ProductVersionMajor	Product major version.
ProductVersionMinor	Product minor version.
ProductVersionBuild	Product build version.
ProductVersionRev	Product revision version.
FileVersionMajor	File major version.
FileVersionMinor	File minor version.
FileVersionBuild	File build version.
FileVersionRev	File revision version.
ProductName	Name of the ActiveX control.
CompanyName	Company name of the ActiveX control.
IsWow64Process	Indicates if loaded in WOW64.
PageUrlZone	Security Zone of the site that the ActiveX control is loaded in.
VersionCheckEnabled	Indicates if blocking of out-of-date ActiveX controls policy is enabled.
State	Indicates if this control was blocked because it is out of date.
Reason	Reason for blocking or allowing the control.
BlockListVersion	Version of the out-of-date ActiveX blocklist.
IsAuditModeEnabled	Indicates if ActiveX control logging policy is enabled in Internet Explorer.
EPMCompatible	Is the control EPM compatible.
Microsoft.Web.Platform.VersionManagementIntranet (Intranet + Trusted + Local File Zone)	These events provide data on a user's Internet Explorer activity.
AbsoluteUri	URL of the site that loaded this ActiveX control. Query string is removed from URL.
Entry	The GUID of the control being logged.
FileName	Name of the file being loaded.
ProductVersionMajor	Product major version.
ProductVersionMinor	Product minor version.
ProductVersionBuild	Product build version.

ProductVersionRev	Product revision version.
FileVersionMajor	File major version.
FileVersionMinor	File minor version.
FileVersionBuild	File build version.
FileVersionRev	File revision version.
ProductName	Name of the ActiveX control.
CompanyName	Company name of the ActiveX control.
IsWow64Process	Indicates if loaded in WOW64.
PageUrlZone	Security Zone of the site that the ActiveX control is loaded in.
VersionCheckEnabled	Indicates if blocking of out-of-date ActiveX controls policy is enabled.
State	Indicates if this control was blocked because it is out of date.
Reason	Reason for blocking or allowing the control.
BlockListVersion	Version of the out-of-date ActiveX blocklist.
IsAuditModeEnabled	Indicates if ActiveX control logging policy is enabled in Internet Explorer.
EPMCompatible	Is the control EPM compatible.