

Device Query, Device Inventory And..... ☺

Your ultimate troubleshooting and reporting tools





Thank you Sponsors



Gold



RECAST SOFTWARE

Silver



Technical Partners



About “Rudy Ooms”





About “Joost Glijsteen”



Website
JoostGlijsteen.com

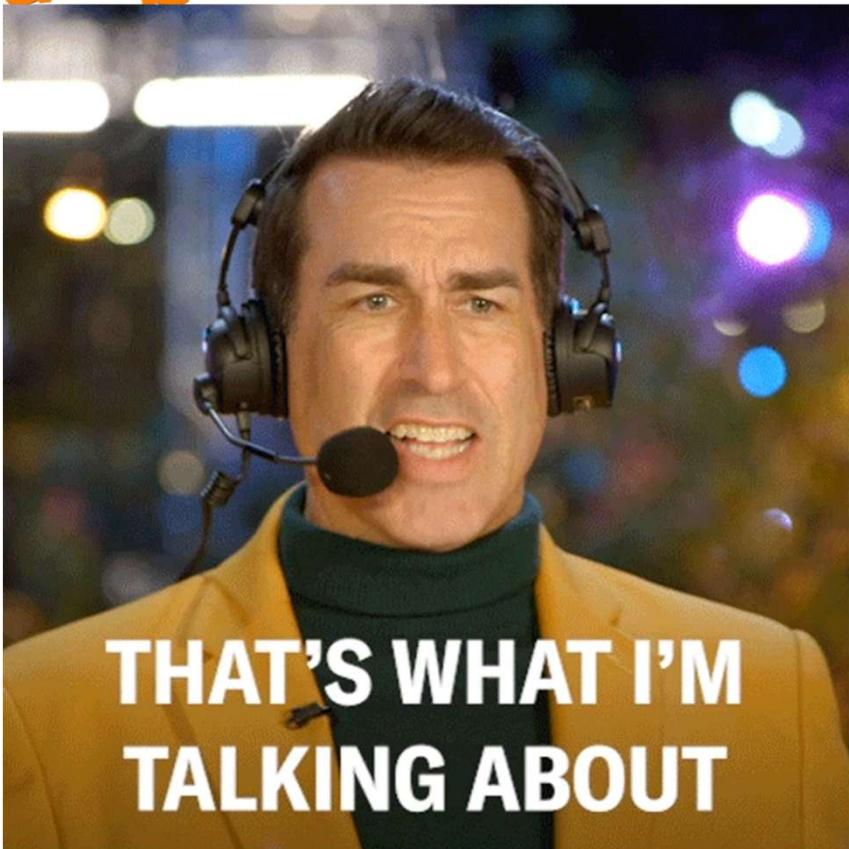


Hobbies
Workplace Ninja NL organization
Blogging
Sneakers

PinkElephant
Solution Lead Workplace



Agenda



1. Introduction
 2. Device Query
 3. Device Inventory
 4. The Device Inventory Agent
 5. The Technical Flow
 6. What's coming!
 7. Comparision
 8. Conclusion
-



1. Introduction

Device Query AND Device Inventory!

The screenshot shows a laptop screen with a "Properties" sidebar on the left containing various device categories like BiosInfo, Certificate, Cpu, DiskDrive, EncryptableVol..., FileInfo, LocalGroup, LocalUserAccou..., and LogicalDrive. The main area has a search bar at the top with "Run" and "Clear input" buttons. Below the search bar is a code editor window showing the following query:

```
1 Cpu  
2 | project ProcessorId, ProcessorType, Architecture , Model
```

At the bottom, there are "Get started" and "Results" tabs, with "Results" being active. A status message "Running query..." is displayed.

The screenshot shows a laptop screen with a "Search" bar at the top. Below it is a table titled "OS Version" with the following data:

Os Name	Architecture	Build Version	Install Date Time
Microsoft Windows 11 Enterprise	64-bit	22631	09/07/2024, 06:09:35 PM

On the left side of the table, there is a list of system components: Battery, Bios Info, Cpu, Disk Drive, Encryptable Volume, Logical Drive, Memory Info, Network Adapter, and OS Version. The "OS Version" item is currently selected.



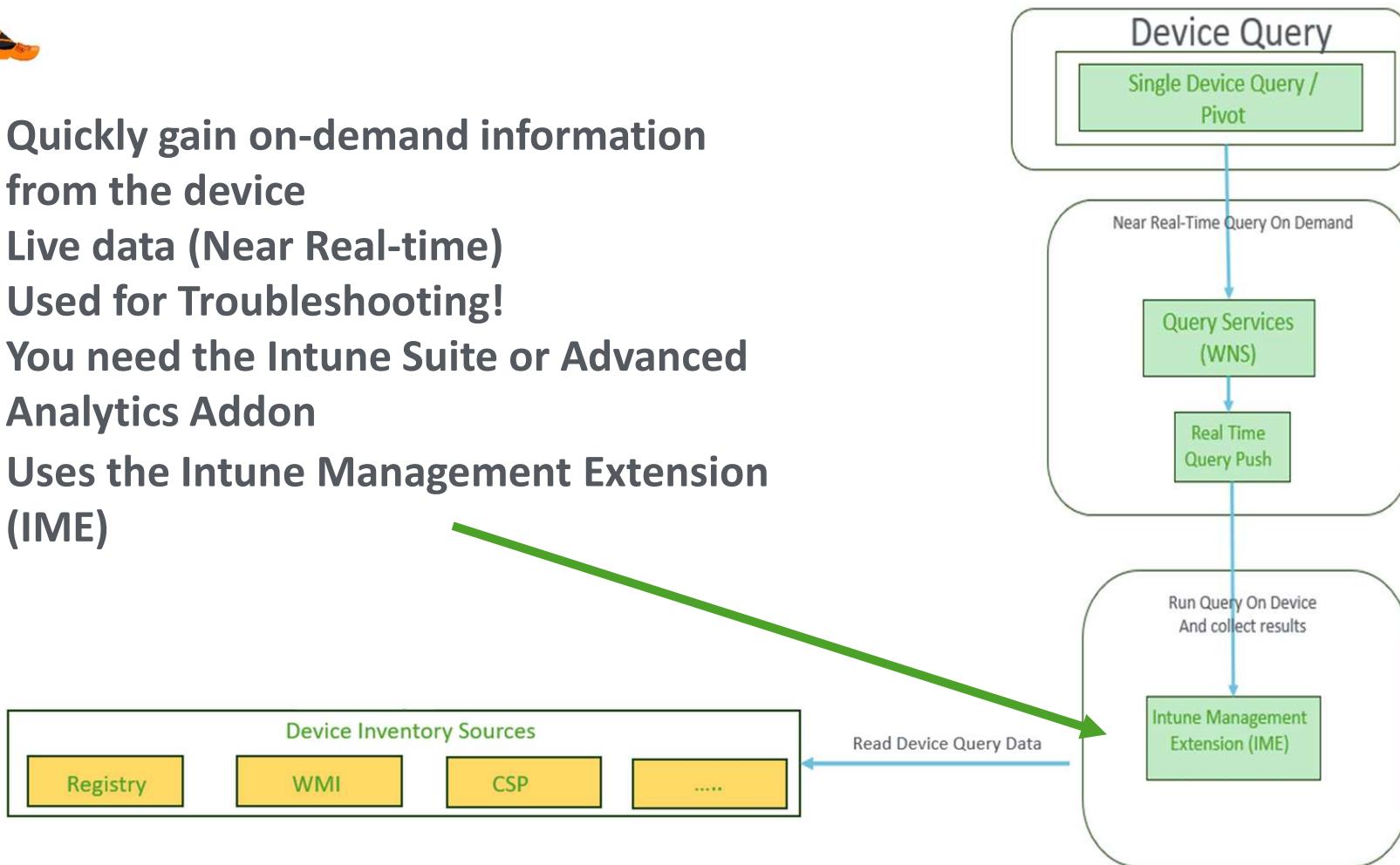
Device Query





Device Query Key Capabilities

- Quickly gain on-demand information from the device
- Live data (Near Real-time)
- Used for Troubleshooting!
- You need the Intune Suite or Advanced Analytics Addon
- Uses the Intune Management Extension (IME)





It's very simple to use!

The screenshot shows the Microsoft Intune Device query interface. On the left, there's a sidebar with various management options like Overview, Manage, Properties, Monitor, Resource explorer (preview), Hardware, Discovered apps, Device compliance, Device configuration, App configuration, Local admin password, Recovery keys, User experience, Device diagnostics, Group membership, Managed Apps, Filter evaluation, Enrollment, and Remediations (preview). The 'Device query' option is highlighted with a blue border.

The main area is titled 'Device query' and contains a 'Properties' section with a search bar and a tree view of device properties. A red box highlights the 'Certificate' node under 'Properties'. To the right, a query editor window is open with the following PowerShell-like query:

```
1 Certificate
2 | where ValidToDateTime < now()
3 | project ValidToDateTime, SubjectName, CommonName
```

A red arrow points from the text 'Query Input' to this query editor window. Below it, a green arrow points from the text 'Query Results' to a table displaying the results of the query. The table has columns for ValidToDateTime, SubjectName, and CommonName, listing several certificates including Verisign, Microsoft, and Thawte.

ValidToDateTime	SubjectName	CommonName
2016-10-24T23:59:59Z	OU=www.verisign....	www.verisign.com/CPS Incorp.by Ref. LIABILITY I
2002-12-31T07:00:00Z	CN=Microsoft Win...	Microsoft Windows Hardware Compatibility
2021-05-09T23:28:13Z	CN=Microsoft Roo...	Microsoft Root Certificate Authority
2020-12-31T23:59:59Z	CN=Thawte Timest...	Thawte Timestamping CA
2020-12-31T07:00:00Z	CN=Microsoft Roo...	Microsoft Root Authority



How good is your KQL?



- Device Query uses KQL (Kusto Query Language)
- Kusto Query Language is used in Log Analytics, Sentinel, and now also for Device Query
- Used for exploring the data and discovering patterns
- KQL operators are sequenced by using a pipe (|)
- Use Copilot to create the KQL queries for you!



```
▶ Run ✖ Clear input ✖ Cancel  
  
1 Certificate  
2 | where ValidToDateTime < now()  
3 | project ValidToDateTime, SubjectName, CommonName
```

Demo Time and let's look at some KQL examples!!!

Search X <>

Properties

Overview

Manage

Properties

Monitor

Resource explorer (preview)

Hardware

Discovered apps

Device compliance

Device configuration

App configuration

Local admin password

Recovery keys

User experience

Device diagnostics

Group membership

Managed Apps

Filter evaluation

Enrollment

Remediations (preview)

Device query

Properties

Search

BiosInfo

Certificate

Cpu

DiskDrive

EncryptableVolume

FileInfo

LocalGroup

LocalUserAccount

LogicalDrive

MemoryInfo

OsVersion

Process

SystemEnclosure

SystemInfo

Tpm

WindowsAppCrash...

WindowsDriver

WindowsEvent

WindowsQfe

Run Clear input Cancel

1

Get started Results

Query the properties of this device on demand, based on device, app, and management properties. Query using the properties provided, or write a custom query using Kusto Query Language (KQL). [Learn more about Device query](#)

Prerequisites

To use Device query in your tenant, you must have a license that includes Microsoft Intune Advanced Analytics. To use Device query on a device, the device must be enrolled in Endpoint Analytics. [Learn more about Device query prerequisites](#)

Kusto database

[Learn about the Kusto query language](#)

[Learn more about entities you can query](#)

[Learn common operators for Kusto Query Language](#)



How Device Query Works



So what happens when we click on “Run”?

How Device Query Works

“Running” the Query

Device query ...

Properties



Search

Run

Clear input

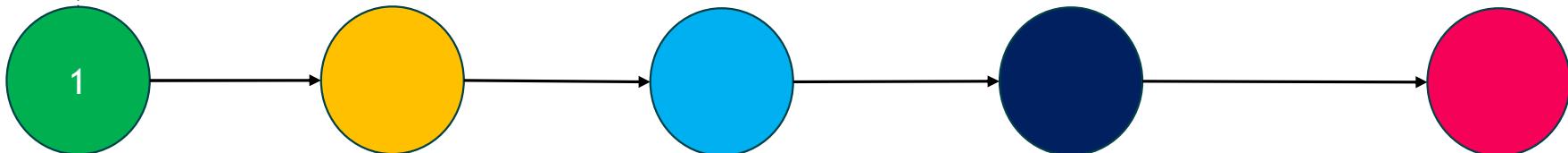
Cancel

BiosInfo

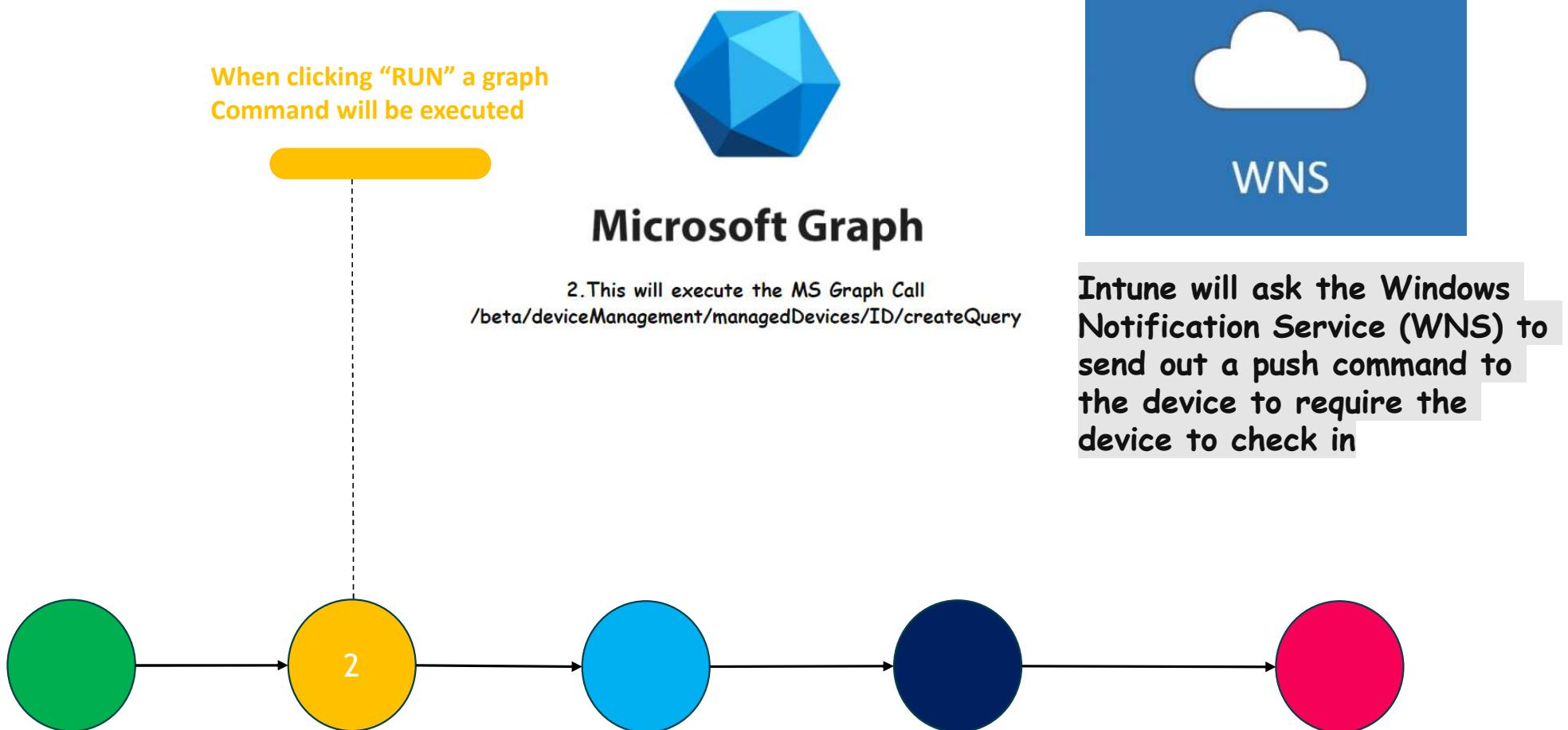
Certificate

1 LocalUserAccount

2 | project UserId, Username, HomeDirectory, WindowsSid



How Device Query Works

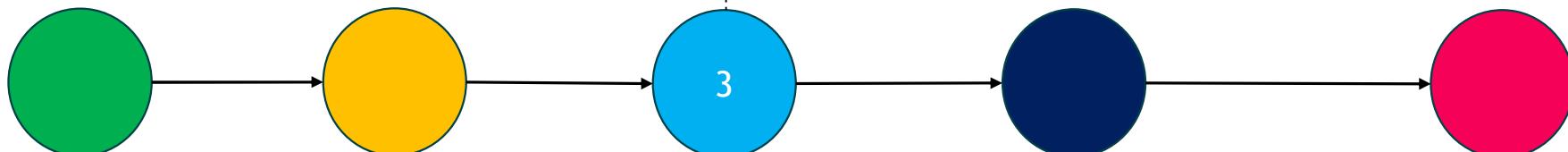


How Device Query Works

The IME Receives the push with the notificationintent. From there on, the device would check in to the service and would retrieve the base64 encoded device query

Query retrieved for user 389de89a-faf6-404d-969e-00facfcdae9a in session 1, raw payload t

Retrieves the base64 encoded query Response from Intune = { "odata.metadata": "https://f... .SideCarGenericPolicy::get_PolicyPayload()" }
- getPolicyPayloadResponse=(null)
- Key=8016ba22-321a-474a-bea6-6dc9b5ffd15e
- odata.id=urn:StatelessSideCarGatewayService/SideCarGatewaySessions(guid'8016ba22-321a-474a-bea6-6dc9b5ffd15e')
- odata.metadata=https://fef.amsb0302.manage.microsoft.com/SideCar/StatelessSideCarGatewayService/\$metadata#SideCarGatewaySessions/@Element
- RequestContentType=GetSideCarGenericPolicies
- RequestPayload=
- ResponseContent-type="application/json; charset=utf-8"/>
- ResponsePayload=[{"PolicyType":13,"PolicyPayload":{"("QueryId")":"8a5ddbf2-9570-4367-bfb3-acb5f5065a21","("QueryAsString")":"Yml0bG9ja2VyX2luZm8NCn"}]
SessionId=8016ba22-321a-474a-bea6-6dc9b5ffd15e



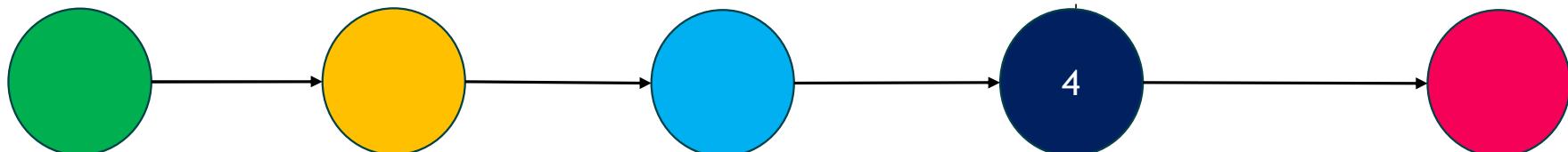
How Device Query Works

```
.IntuneWindowsAgent.AgentCommon.DeviceQueryResult RunQuery()
```

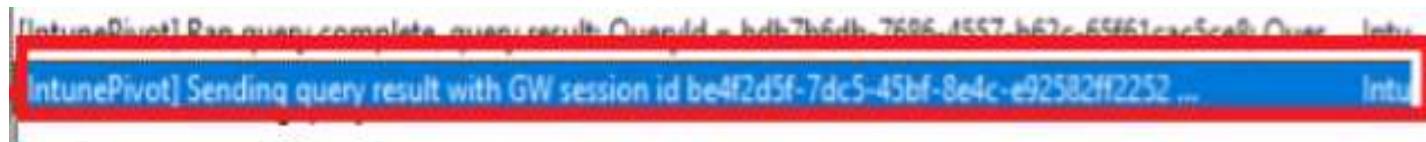


After receiving the Device Query, the IME will ask the Windows Management Instrumentation to gather the requested data

IME Executing Query /
Run Query



How Device Query Works



```
OPTIONS https://graph.microsoft.com/beta/deviceManagement/managedDevices/7f49b96f-51db-4600-b932-7cba2e2481fc/queryResults/  
Host: graph.microsoft.com  
Connection: keep-alive  
Accept: */*  
Access-Control-Request-Method: GET
```



The Query results will show up in the portal so we could take a look at it

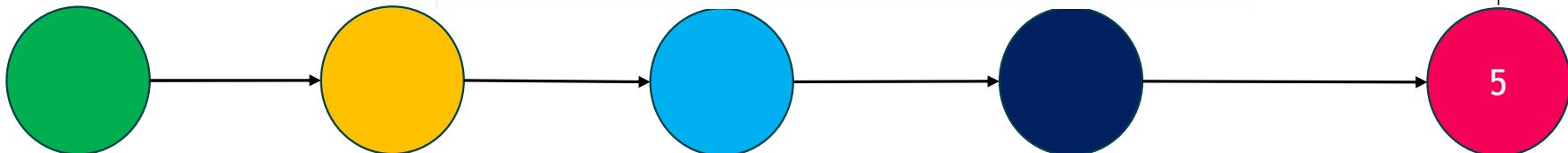


Userid	Username	UserDescription	HomeDirectory	WindowsSid
500	BUILTIN\Administrators	Built-in account for local administrator access	\$-1-5-21-382041948-2858433630-3437510959-500	S-1-5-21-382041948-2858433630-3437510959-500
503	DefaultAccount	A user account managed by Intune	\$-1-5-21-382041948-2858433630-3437510959-503	S-1-5-21-382041948-2858433630-3437510959-503
501	Guest	Built-in account for remote users	\$-1-5-21-382041948-2858433630-3437510959-501	S-1-5-21-382041948-2858433630-3437510959-501
504	WDAGUtilityAccount	A user account managed by Windows Defender Antivirus	\$-1-5-21-382041948-2858433630-3437510959-504	S-1-5-21-382041948-2858433630-3437510959-504



Microsoft Intune

IME sends the query Results back to the service

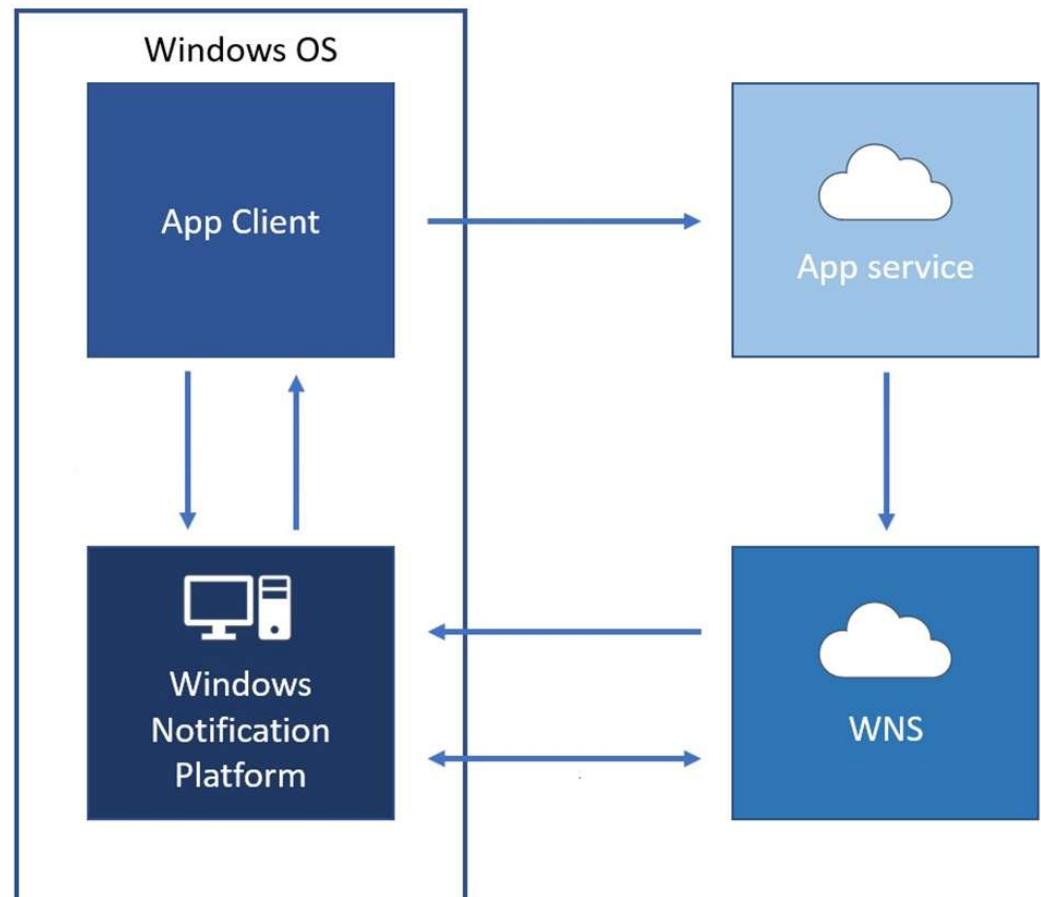


5



Windows Notification Service

- Device Query relies on the Windows Notification Service
- Windows Notification Service is used to deliver “push notifications” to the device!
- Don’t Block WNS in your network!
(SSL Inspection)





An error occurred

| Device query ...

Properties

Search

- ✓ BiosInfo
- ✓ Certificate
- ✓ Cpu
- ✓ DiskDrive
- ✓ EncryptableVolume
- ✓ FileInfo
- ✓ LocalGroup
- ✓ LocalUserAccount
- ✓ LogicalDrive
- ✓ MemoryInfo

Device query ...

Run Clear input Cancel

1 Cpu | take 40

Get started Results

Check your query for syntax errors and run it again.

Properties

Search

- ✓ BiosInfo
- ✓ Certificate
- ✓ Cpu
- ✓ DiskDrive
- ✓ EncryptableVolume
- ✓ FileInfo
- ✓ LocalGroup
- ✓ LocalUserAccount
- ✓ LogicalDrive
- ✓ MemoryInfo
- ✓ OsVersion
- ✓ Process

Device query ...

Run Clear input Cancel

1 OsVersion
2 | project Architecture, BuildVersion

Get started Results

You can only send 15 queries per minute. Wait a minute, and then try again.

Properties

Search

- ✓ BiosInfo
- ✓ Certificate
- ✓ Cpu
- ✓ DiskDrive
- ✓ EncryptableVolume
- ✓ FileInfo
- ✓ LocalGroup
- ✓ LocalUserAccount
- ✓ LogicalDrive
- ✓ MemoryInfo
- ✓ OsVersion
- ✓ Process

Get started Results

An error occurred. If this happens again, contact your IT support.

What If Device Query is not working?

Troubleshooting Device Query



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\IntuneManagementExtension\Notification

Name	Type	Data
(Default)	REG_SZ	(value not set)
ChannelAddress	REG_SZ	https://wns2-am3p.notify.windows.com/?token=AwYAAACAZFaJR9qw7EKeyOA31sYgZqQNMasyZcqevaG...
EnableChannelU...	REG_SZ	True
EnableDeviceAc...	REG_SZ	True
ErrorMessage	REG_SZ	
ExpirationDate	REG_SZ	08/02/2024 07:56:24
FullSyncFrequen...	REG_DWORD	0x000000a8 (168)
WNSChannelLas...	REG_SZ	0

You cannot opt out of cloud notifications (WNS)

MDM Push: Failed to create WNS Push Channel for MDM Push Sessions. Result: (Cloud notifications have been turned off.).

Registry Editor

File Edit View Favorites Help

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\CurrentVersion\PushNotifications

Name	Type	Data
(Default)	REG_SZ	(value not set)
NoCloudApplicationNotification	REG_DWORD	0x00000001 (1)



Troubleshooting Device Query

- Device presence – offline/online
- Did WNS notification reach client device?
- Need to get client device IME log, and check if a log line like this happened after clicking run for the query:
[DeviceAction] PushNotification workload thread start with id: 7c704305-8958-463a-98ec-ff96d273c6c1, intent: IntunePivot , version: 1, arguments: System.Collections.Generic.Dictionary`2[System.String,System.String]
- If WNS notification didn't arrive, check client IME log history up to previous 8 hours to confirm if WNS channel URI registered correctly. If yes, there would be a log line like this:
[GetChannelUriFromWns] Registered channel URI: https://wns2-bn3p.notify.windows.com/?token=AwYAAADyKvIYc1IA%2bBQFHd9fGNgnUpgkRIHbDOIFR2Lx2fCmhSat%2ffOLccJaxORBrkqWfFpf4978Cd%2fZC%2b8vR40JZgNQUX7m09lwVeo83iFJccu32gbHlmd36MfPp20XpmqjpqY%3d
- A typical failure channel URI registering error log would be like this:
"fail to register push notification channel, exception is System.Exception: Cloud notifications have been turned off. (Exception from HRESULT: 0x803E0109)"
- The above error would happen if client device opt out from WNS notification because of MDM policy etc. To fix this issue, Admin need to figure out if any such policy configured and applied to this device. If yes, Admin need to exempt the device from the policy.
- For new enrolled device, it would take up to 1 hour for WNS notification channel info to populate to WNS service, so that an up to 1 hour delay for new enrolled device is expected.
- For device with WNS channel URI registration failure, even after the WNS notification opt out issue solved (client device exempted from WNS notification opt out policy), it will take up to 8 hours for SidecarAgent to retry register channel URI, so that up to 8 hours of the delay after policy switch would be expected. Reboot the machine or restart SidecarAgent could speed the recovery up

Device Inventory





Key capabilities

- Intune Core (plan 1) feature!
- Configure from available entities in Inventory Catalog.
- Expansion of available entities.
- Reporting on configuration.
- Per device view of collected Inventory.
- Visibility into data freshness
- Efficient collection mechanisms to minimize overhead on network and device compute
- Low latency pipeline.
- It's extensible (let Rudy explain this)

The image contains two screenshots of the Microsoft Intune admin center. The top screenshot shows the 'Properties picker' interface, where users can select properties from a catalog to add to a profile. The selected properties are listed on the right: Address Width (Cpu), Architecture (Cpu), Cpu Status (Cpu), and Manufacturer (Cpu). The bottom screenshot shows the 'Resource explorer' for a specific device (DESKTOP-FEKSTB7). The left sidebar lists categories like Overview, Favorites, Manage, and Monitor. The main pane displays resource details: Battery, Bios Info, Cpu, Disk Drive, and Encryptable Volume.



Device Harvesting

The Agent is Extensible and declarative

- Flexible Setup: The system is designed to handle different Properties/entities without needing extra coding.
- Automatic Adaptation: New configured Properties/Entities are automatically added to the agent

- No Manual Updates: You don't have to manually update the code for new queries; the system does it for you.

```
"\\\".\\|ROOT\\MicrosoftDeviceManagement_Extensibility_Inventory")
```

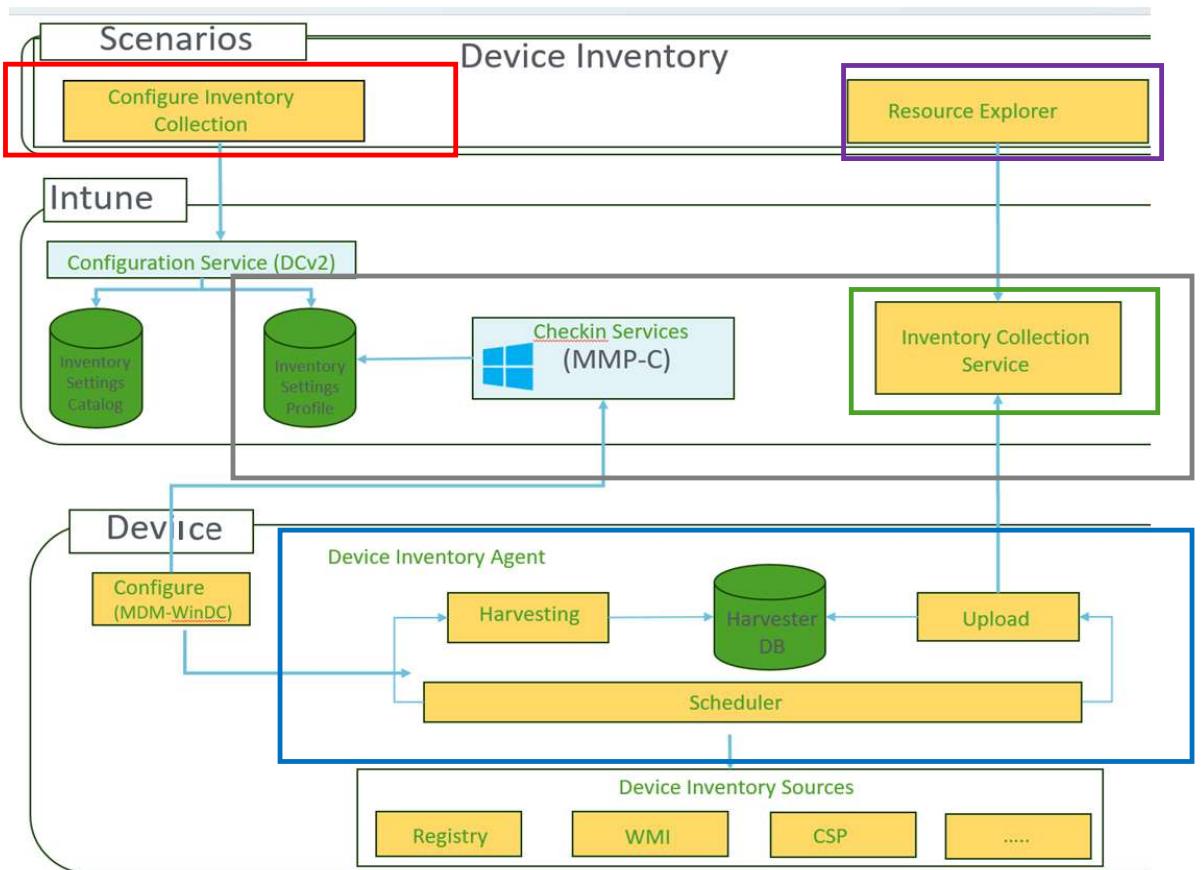




Device Inventory Global Overview

- We need to Configure a Policy to start the “harvesting”
- Policy Delivery (MMP-C)
- Data is stored in the service
- Resource Explorer is used to look At the data stored in the service
- Uses own Device Inventory agent

We will guide you through every step!

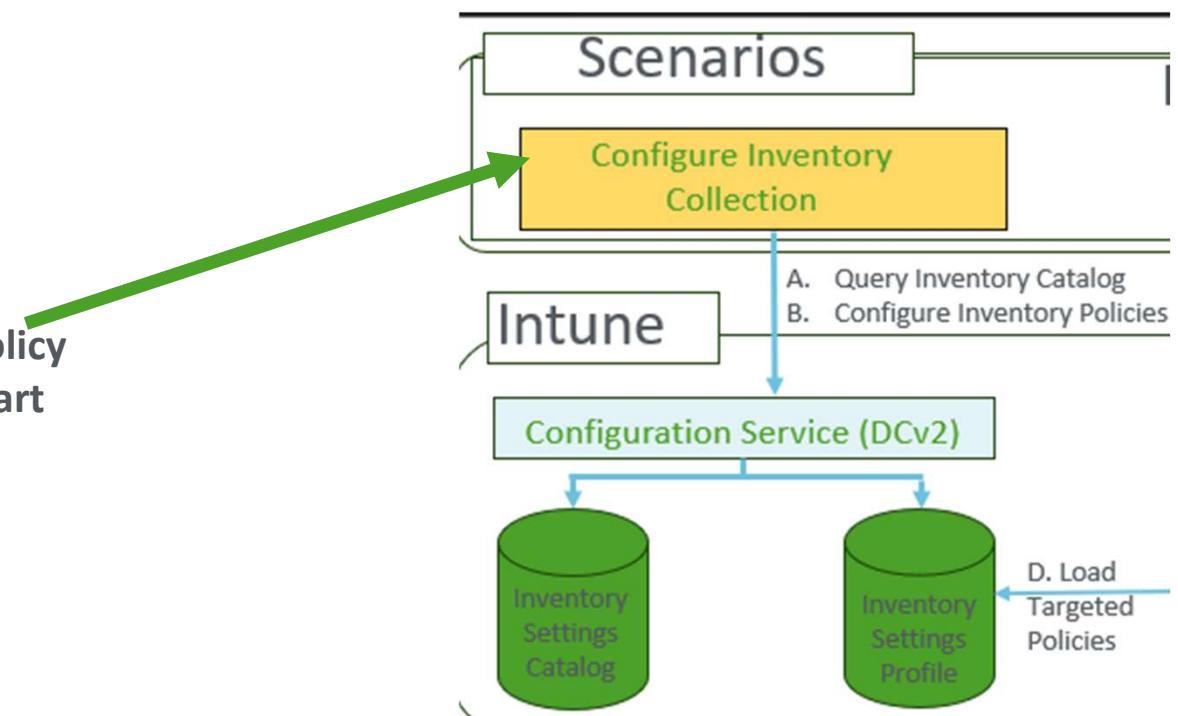




Configuration

We need to configure a new Policy called: Properties Catalog to start the Harvesting of the data

Let's do so!!!

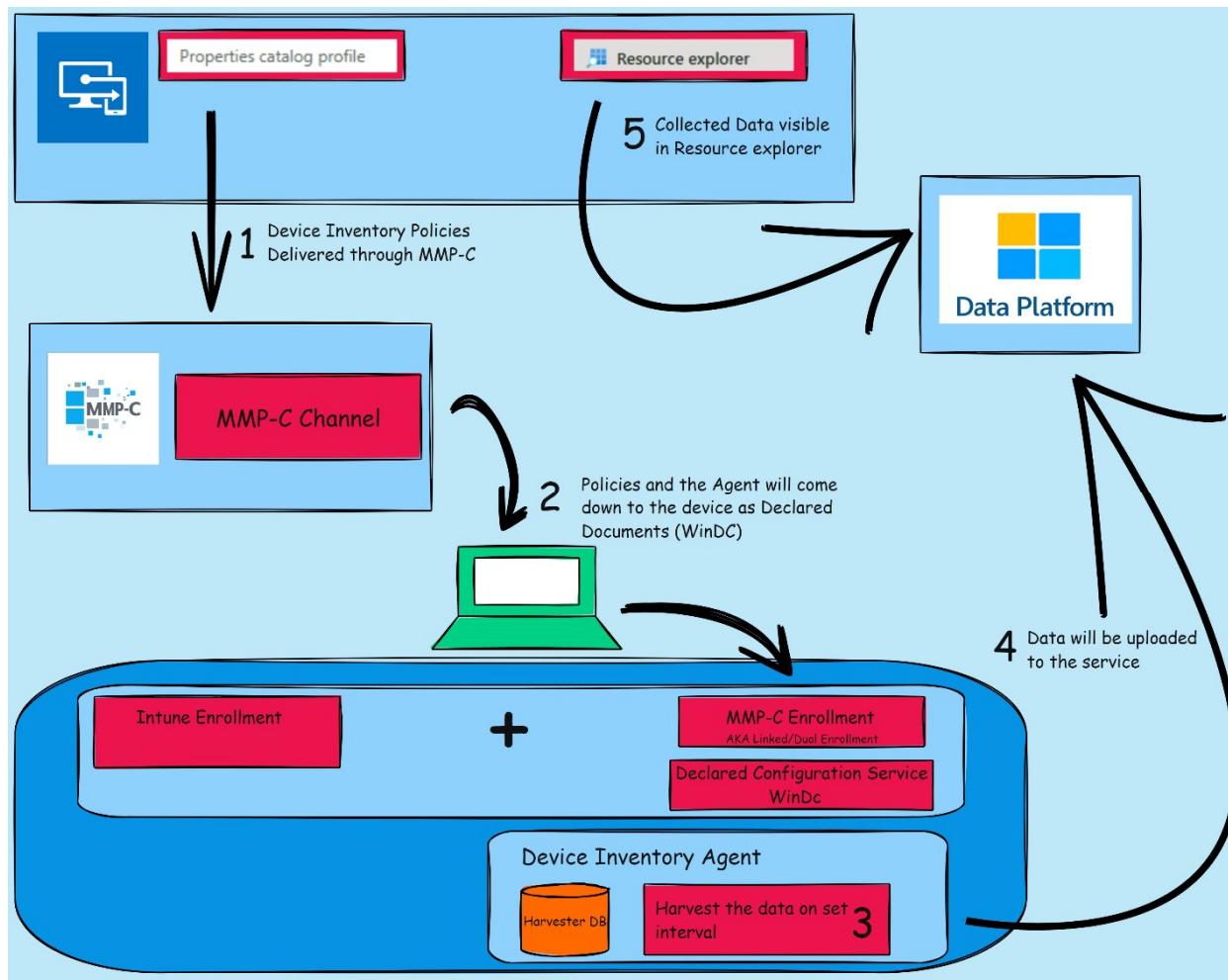


Configure the Properties Catalog

The screenshot shows the Microsoft Intune admin center interface. The left sidebar navigation includes Home, Dashboard, All services, Devices, Apps, Endpoint security, Reports, Users, Groups, Tenant administration, Troubleshooting + support, and several sections under Manage devices: Configuration, Compliance, Conditional access, Scripts and remediations, Group Policy analytics, eSIM cellular profiles (preview), Policy sets, Device categories, Partner portals, Manage updates (Windows updates, Apple updates, Android FOTA deployments), and Organize devices (Device clean-up rules). The main content area is titled "Devices | Configuration" and displays a list of 14 policies. The columns are: Policy name, Platform, Policy type, Last modified, and Scope tags. The policies listed are:

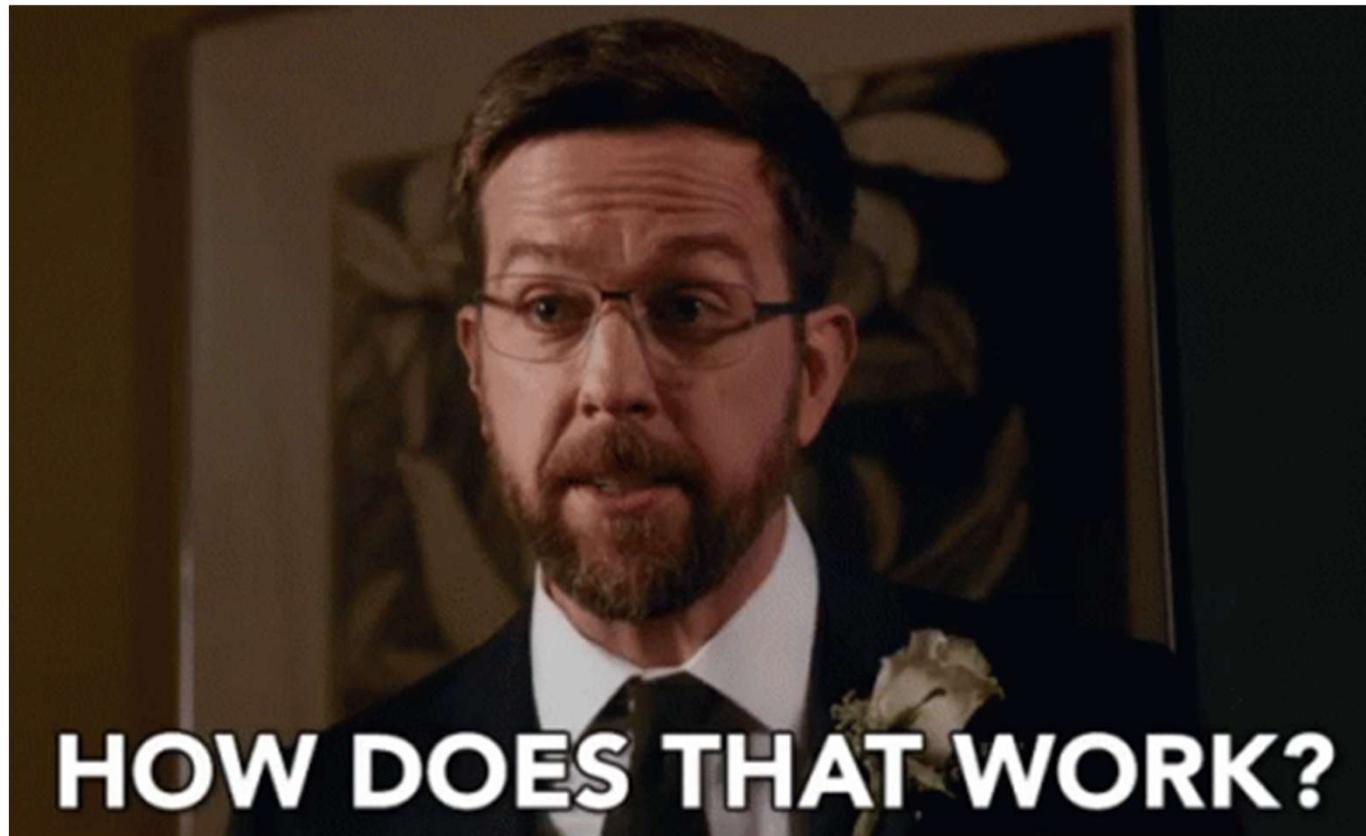
Policy name	Platform	Policy type	Last modified	Scope tags
auto trigger policy	Windows 10 and later	Properties catalog	8/29/2024, 1:17:06 PM	1 assigned
Ignacio Test V4	Windows 10 and later	Properties catalog	8/23/2024, 2:07:37 PM	1 assigned
jasbirTestProfile	Windows 10 and later	Properties catalog	7/9/2024, 3:22:58 PM	1 assigned
Log_test	Windows 10 and later	Properties catalog	8/7/2024, 12:00:01 PM	1 assigned
oldSettings	Windows 10 and later	Properties catalog	7/2/2024, 6:13:32 PM	1 assigned
Policy for Co-Man	Windows 10 and later	Properties catalog	6/19/2024, 3:52:51 PM	1 assigned
sdfsdfsdf	Windows 10 and later	Properties catalog	6/25/2024, 2:46:05 PM	1 assigned
test	Windows 10 and later	Properties catalog	7/15/2024, 12:49:04 PM	1 assigned
Test policy creation platform	Windows 10 and later	Properties catalog	8/15/2024, 11:10:10 AM	1 assigned
Test Profile	Windows 10 and later	Properties catalog	8/30/2024, 3:43:47 AM	1 assigned
Test SettingsCatalog	Windows 10 and later	Settings catalog	6/14/2024, 2:40:47 PM	1 assigned
testInventory	Windows 10 and later	Properties catalog	7/25/2024, 10:06:26 AM	1 assigned
Vinyas_Log_Test	Windows 10 and later	Properties catalog	7/24/2024, 11:56:35 AM	1 assigned
Vinyas_test_profile	Windows 10 and later	Properties catalog	8/30/2024, 3:39:03 AM	1 assigned

Policy Delivery





Inventory Agent



How does the data end up in Intune?

Deepdive Device Inventory Agent

Configuring The new Harvest Policy (Properties)

1.Create the Properties Catalog

Platform

Windows 10 and later

Profile type

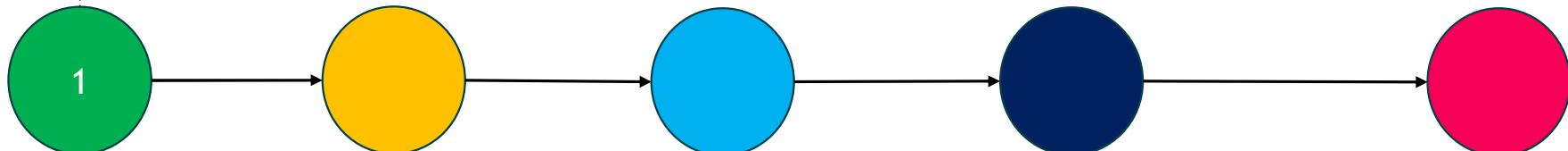
Properties catalog

2. Define all the Properties (Entities) we want and save the policy

Properties picker

The screenshot shows a 'Properties picker' window. On the left, a list of entities is displayed with arrows indicating they can be selected. On the right, a panel titled 'Selected properties (9)' lists the chosen entities with their types and delete icons.

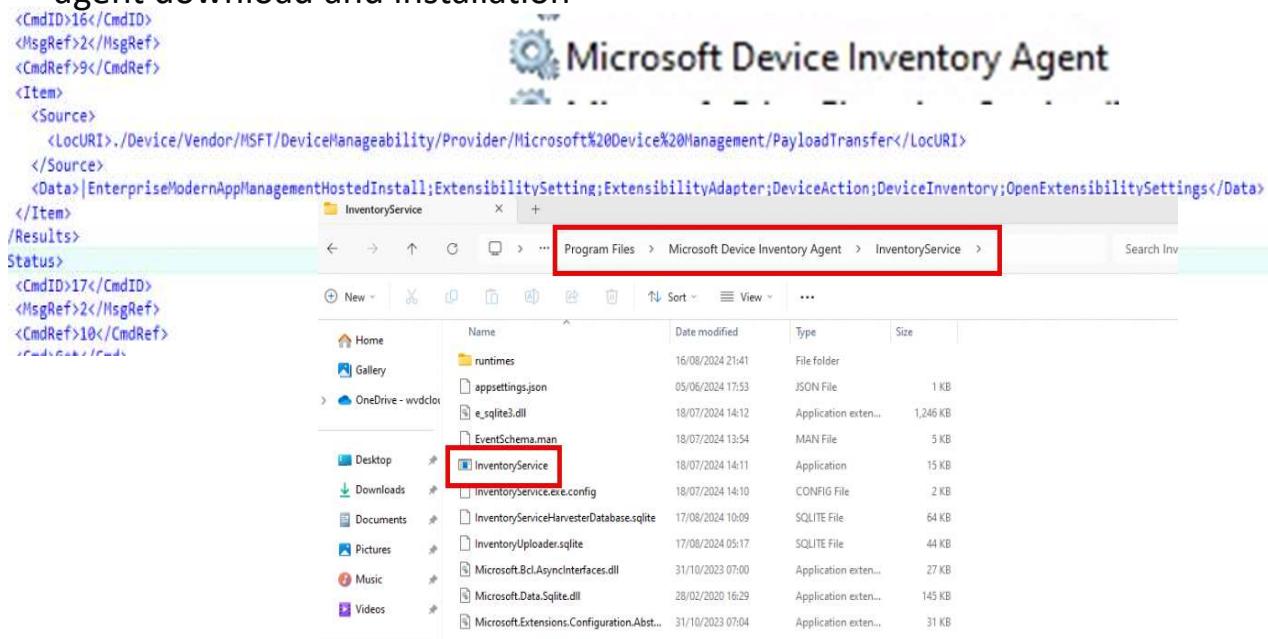
Entity	Type	Action
Time Zone	Time	Delete
Activated	Tpm	Delete
Spec Version	Tpm	Delete
Physical Presence Version	Tpm	Delete
Encryptable Volume		
Logical Drive		
Memory Info		
Network Adapter		
OS Version		
System Enclosure		



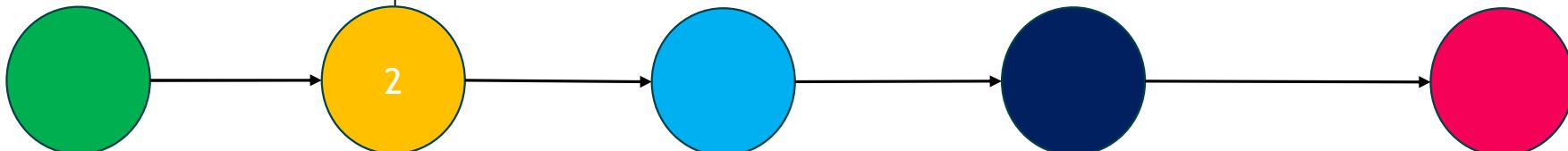
Deepdive Device Inventory Agent

A CSP will be sent to the device to initiate the agent download and installation

The Device Inventory Agent will be installed



The Device Inventory Agent will show up in the program files folder.



Deepdive Device Inventory Agent

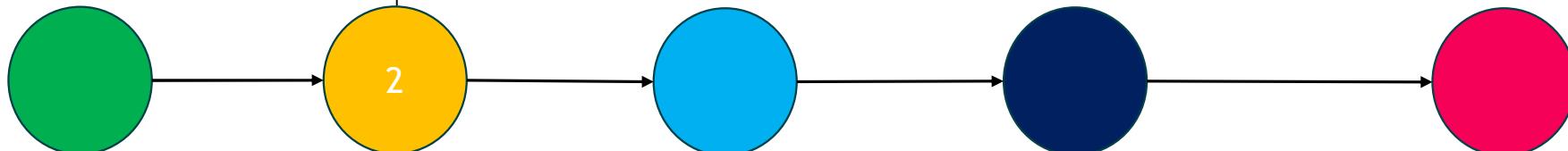
The Device Inventory
Agent will be installed

The screenshot shows the Windows Registry Editor with a focus on the registry key for the Device Enrollee. The key path is `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\DeviceEnrollee`. The registry table on the right lists several keys:

Name	Type	Data
(Default)	REG_SZ	(value not set)
DiscoveryEndpoint	REG_SZ	https://discovery.dm.microsoft.com/EnrollmentC...
EnrollStatus	REG_DWORD	0x00000003 (3)
LastError	REG_DWORD	0x00000000 (0)
LinkedEnrollmentId	REG_SZ	A39C1AD4-3D46-4A7D-8AE5-774020D8AAEB
MMPCLocked	REG_DWORD	0x00000001 (1)

The registry table on the left shows the structure of the registry key, with the `FirstSync\LinkedEnrollment` key highlighted with a red box.

Troubleshooting Tip! If somehow (SSL Inspection / Proxy on dm.Microsoft.com) the inventory agent or policies don't show up. Please check if the device has a linkedenrollment and is locked/connected with MMP-C as device inventory relies on MMP-C

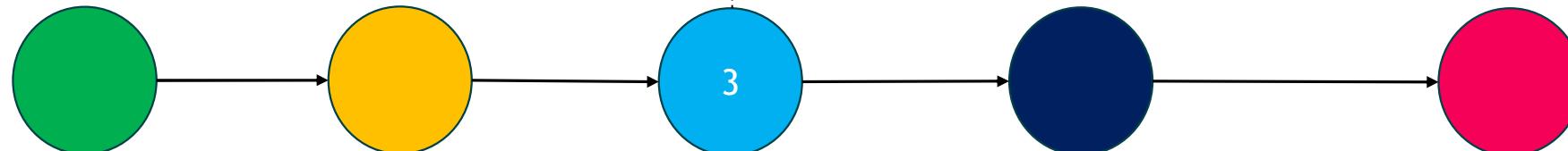


Deepdive Device Inventory Agent

The Device Inventory Policies (What we want to harvest) will come down through the MMP-C channel as declared configuration documents.

These documents will be handled by WinDc (Declared Configuration Service)

```
Service name: dcsvc
Display name: Declared Configuration(DC) service
Description: Process Declared Configuration documents received from MDM and other channels and perform configurations on device
Path to executable:
C:\WINDOWS\system32\svchost.exe -k netsvcs -p
```

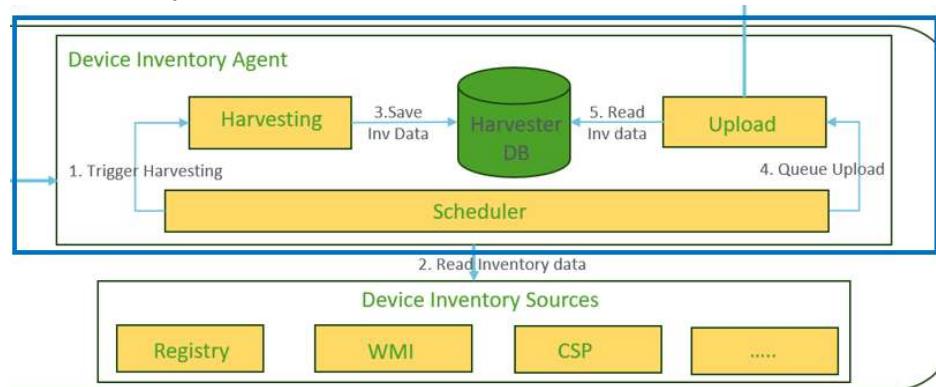


With the agent on the device, the Inventory policies also need to be sent to the device

```
<CmdRef>12</CmdRef>
<Item>
<Source>
<LocURI>./Device/Vendor/MSFT/DeclaredConfiguration/Host/Inventory/Results/8e3b60d3-9a8c-34cb-4809-24729372b0ce/Document
<Dest>
<ReflectedProperties>&lt;Property name="MeId"&gt;MeType;MeVersion;IntentId type="chr"&gt;f48b6647-a1f7-403d-8590-9f8c90105a51;15;1</ReflectedProperties>&lt;Property name="NamespaceName" type="chr"&gt;Windows:Device:Inventory&lt;/Property>&lt;Property name="IntentPriority" type="chr"&gt;10&lt;/Property name="ConfigDocumentType" type="chr"&gt;3&lt;/Property>&lt;ReflectedProperties>&lt;Property name="DSC namespace" value="root/MicrosoftDeviceManagement_Extensibility_Inventory" class="InventoryAdaptor" status="200" state="80"&gt;&lt;Key name="DocumentID"&gt;dc7cbe48-2131-ec8d-00d5-3f7a8846957&lt;/Key&gt;&lt;Key name="Version"&gt;CC3D8CB381364FAF89602299C92F2EB0792C70E3BF02C9E5A64BD30B1191D86C&lt;/Key&gt;&lt;Value name="ExtendedP Value"&gt;&lt;Value name="InventoryPayload"&gt;[{"PolicyId": "ed51db1b-f336-4768-8fa3-d73c7ed41d4c", "ReportingId": "791a7d dc10-3b959dda79d5", "SettingId": "windows_deviceinventory_encryptablevolume_volumeid_encryptionmethod"}, {"Data": {"AdapterStatus": 0, "HResult": 1, "ErrorDescription": "Data not yet available"}, "HarvesterStatus": {"Status": 0, "HResult": -1, "ErrorDescription": "Data not yet available"}]}&lt;/Value&gt;&lt;DSC&gt;&lt;DSC namespace="root/MicrosoftDeviceManagement_Extensibility_Inventory" class="InventoryAdaptor" status="200" state="80"&gt;&lt;Key name="DocumentID"&gt;dc7cbe48-2131-ec8d-00d5-3f7a8846957&lt;/Key&gt;&lt;Key name="Version"&gt;CC3D8CB381364FAF89602299C92F2EB0792C70E3BF02C9E5A64BD30B1191D86C&lt;/Key&gt;&lt;Value name="ExtendedP Value"&gt;&lt;Value name="InventoryPayload"&gt;[{"PolicyId": "ed51db1b-f336-4768-8fa3-d73c7ed41d4c", "ReportingId": "791a7d dc10-3b959dda79d5", "SettingId": "windows_deviceinventory_encryptablevolume_volumeid_encryptionmethod"}, {"Data": {"AdapterStatus": 0, "HResult": 1, "ErrorDescription": "Data not yet available"}, "HarvesterStatus": {"Status": 0, "HResult": -1, "ErrorDescription": "Data not yet available"}]}&lt;/Value&gt;&lt;DSC&gt;
```

Deepdive Device Inventory Agent

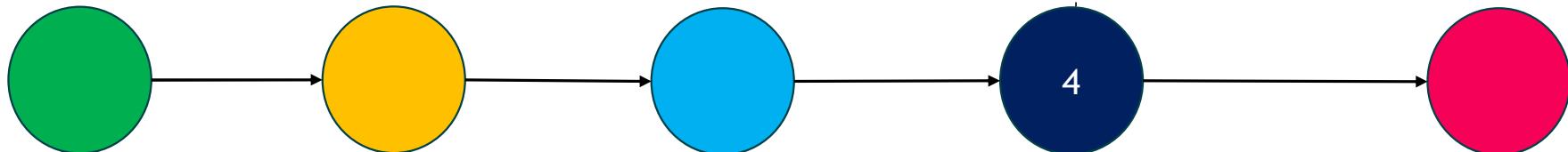
The Device inventory agent harvest the data and upload its to the service. With the data uploaded we can look at it with the resource explorer.



Who wants to see a SUB deep dive on this one?



The agent will start the harvest after Receiving the policies.



Sub Deep Dive: Device Harvesting

1. The Device Inventory Agent harvests this data from the device on a regular base (240 minutes)

runtimes	24/07/2024 21:45
appsettings.json	06/06/2024 02:52
c.sqlite3.dll	18/07/2024 23:12
EventSchema.man	18/07/2024 22:54
InventoryService	18/07/2024 23:11
InventoryService.exe.config	18/07/2024 23:10
InventoryServiceHarvesterDatabase.sqlite	02/08/2024 10:32
InventoryUploader.sqlite	02/08/2024 11:14



2. When harvesting the data, the agent will check the harvester sqlite database for the configured policies/resources (for example tpm)

PolicyId	SettingId
2bf55cae-0672-459a-b6e4-c565b25fd609	windows_deviceinventory_tpm_singlet



3. The Device Inventory agent will collect the properties using Windows Management Instrumentation (WMI) and will store the output in a temp harvested.txt

InventoryHarvesterEntitySet { InventorySource: DeviceInventoryHarvester, serviceCatalog, Category: DeviceInventory, More: { "InventoryEntitySet" }
New Harvested entry ##### InventoryHarvesterEntitySet { InventorySource: DeviceInventoryHarvester, serviceCatalog, Category: DeviceInventory, More: { "InventoryEntitySet" }



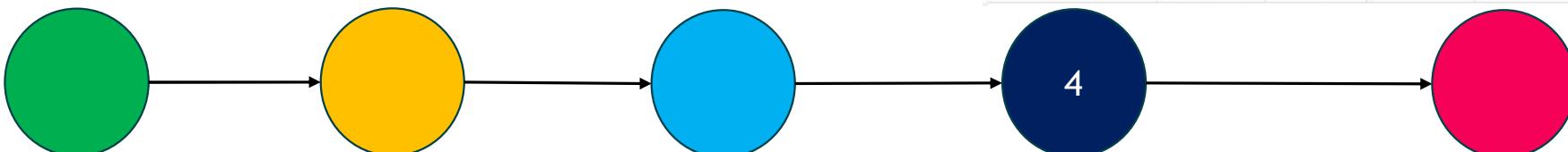
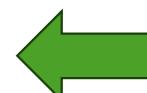
4. Before upload, the harvested data is stored in the Inventory Uploader Database

InventoryUploaderHarvestersShared	Source	Category	EntitySource	EntitySetId	EntityId
DeviceInventoryHarvester	DeviceInventory	ServiceCatalog	Tpm	Singleton	
DeviceInventoryHarvester	DeviceInventory	ServiceCatalog	SystemEnclosure	SerialNumber=6825-9279-8533	
DeviceInventoryHarvester	DeviceInventory	ServiceCatalog	LogicalDrive	DriveIdentifier=C:	
DeviceInventoryHarvester	DeviceInventory	ServiceCatalog	LogicalDrive	DriveIdentifier=D:	

5. The harvested data will be efficiently be sent to the Service once each 24 hours

Request Headers	[RAW]	[Header]
POST /TrafficGateway/TrafficRoutingService/DeviceDataUpload/StatelessDeviceDataUploadService/InventoryDataUpload/Upload HTTP/1.1		
Entity		
Content-Length: 2670		
Content-Type: application/json; charset=utf-8		

Data will be uploaded the service fef.amsub0302.manage.microsoft.com



Deepdive Device Inventory Agent

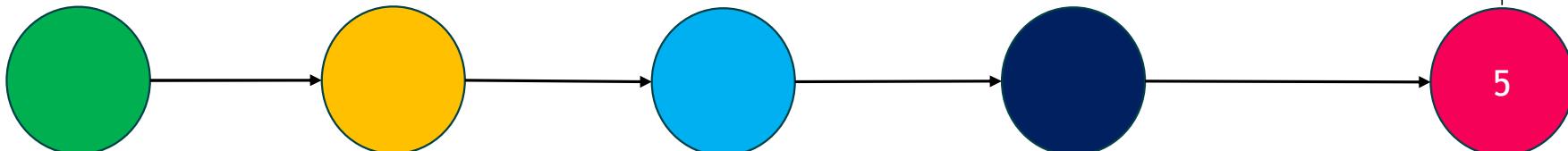
- Resource Explorer will start showing the data after policy has been deployed and the initial harvesting has been executed (give it 24 hours! Ignore the errors)

Architecture ✖ Error 2147749902

CPU Status ✖ Error 2147749902

- The initial harvest will happen in a randomized timeframe.
- No data will be shown if you did not add the property in the properties settings policy
- If Microsoft adds more properties, you need to change the policy manually to add them!

- DEMO TIME!!!



With the inventory data being uploaded to the service, we can now take a look at it with the resource explorer

The screenshot shows the Microsoft Resource Explorer interface for a desktop computer named DESKTOP-14V2Q0D. The main page displays a search bar and a message: "This page allows you to view device information you're co...". On the left, there's a navigation sidebar with links for Overview, Manage, Properties, Monitor (which is expanded), Resource explorer (highlighted with a green box), and Hardware. To the right, there are sections for Battery, Bios Info, and Cpu. A red callout box labeled "Data available in Resource Explorer" points to the Resource explorer link in the sidebar.

5

[Home](#) > [Devices](#)

Devices | All devices

[Search](#) [Refresh](#) [Export](#) [Columns](#) [Bulk device actions](#)

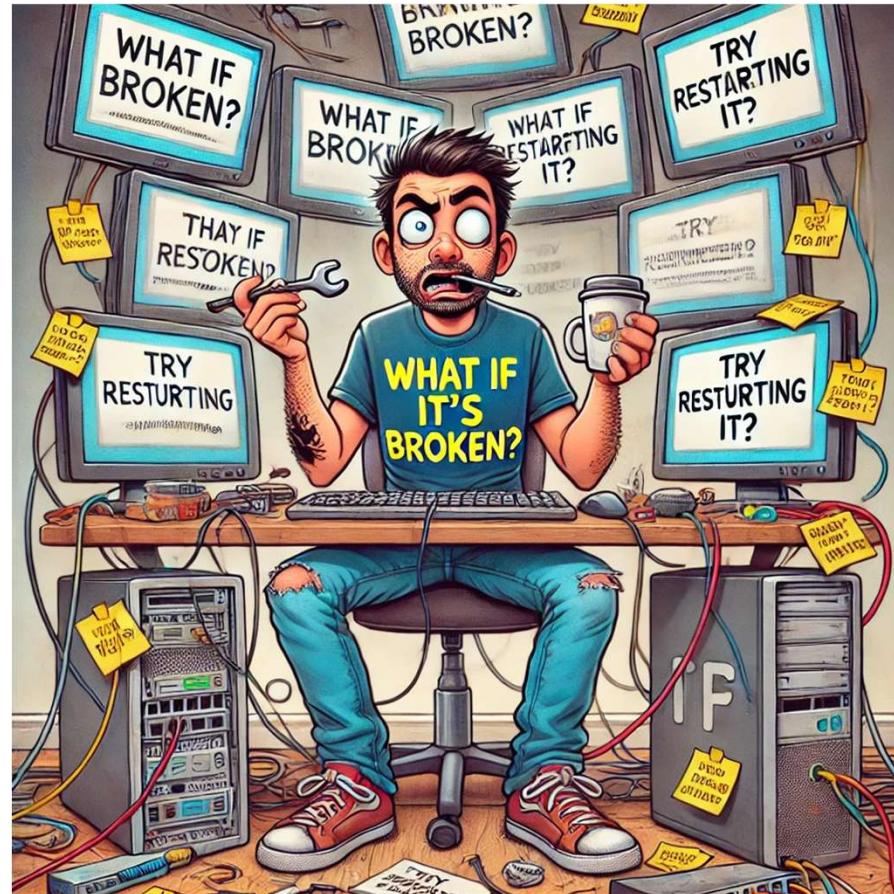
4 devices

[Overview](#) [All devices](#) [Monitor](#)[By platform](#)[Windows](#)[iOS/iPadOS](#)[macOS](#)[Android](#)[Linux](#)[Device onboarding](#)[Windows 365](#)[Enrollment](#)[Manage devices](#)[Configuration](#)[Compliance](#)[Conditional access](#)[Scripts and remediations](#)[Group Policy analytics](#)[eSIM cellular profiles \(preview\)](#)[Policy sets](#)[Device categories](#)[Partner portals](#)[Manage updates](#)[Windows updates](#)[Apple updates](#)[Android FOTA deployments](#)[Organize devices](#)[Device clean-up rules](#)[Search](#) [Add filters](#)

Device name	Managed by	Ownership	Compliance	OS	OS version	Primary user UPN	Last check-in
R61889787	Intune	Corporate	Compliant	Windows	10.0.19045.4780	admin@contoso.onmicro...	09/05/2024, 12:23 PM
R61889788	Intune	Corporate	Compliant	Windows	10.0.22621.4037	admin@contoso.onmicro...	09/05/2024, 12:18 PM
R61899871	Intune	Corporate	Compliant	Windows	10.0.19045.4780	admin@contoso.onmicro...	08/29/2024, 02:49 PM
R61900907	Intune	Corporate	Compliant	Windows	10.0.22621.4037	admin@contoso.onmicro...	08/26/2024, 06:43 AM



Device Inventory Agent “Logs”





Device Inventory Agent “Logs”

at Microsoft.Intune.Inventory.Service.Harvester.Database.SqliteHarvesterDatabase.SaveEventToDb(Event eventToCollect)

5/12/2024 18:38:23 [Exception] Microsoft.Data.Sqlite.SqliteException (0x8004005): SQLite Error 1: 'no such table: InventoryEventCollector'

at Microsoft.Data.Sqlite.SqliteException.ThrowExceptionForRC(Int32 rc, sqlite3 db)

at Microsoft.Data.Sqlite.SqliteCommand.<PrepareAndEnumerateStatements>d__64.MoveNext()

at Microsoft.Data.Sqlite.SqliteCommand.<GetStatements>d__54.MoveNext()

at Microsoft.Data.Sqlite.SqliteDataReader.NextResult()

at Microsoft.Data.Sqlite.SqliteCommand.ExecuteReader(CommandBehavior behavior)

at Microsoft.Data.Sqlite.SqliteCommand.ExecuteNonQuery()

at Microsoft.Intune.Inventory.Service.Harvester.Database.SqliteHarvesterDatabase.SaveEventToDb(Event eventToCollect)

5/12/2024 18:38:23 [Exception] Microsoft.Data.Sqlite.SqliteException (0x8004005): SQLite Error 1: 'no such table: InventoryEventCollector'

at Microsoft.Data.Sqlite.SqliteException.ThrowExceptionForRC(Int32 rc, sqlite3 db)

at Microsoft.Data.Sqlite.SqliteCommand.<PrepareAndEnumerateStatements>d__64.MoveNext()

at Microsoft.Data.Sqlite.SqliteCommand.<GetStatements>d__54.MoveNext()

at Microsoft.Data.Sqlite.SqliteDataReader.NextResult()

at Microsoft.Data.Sqlite.SqliteCommand.ExecuteReader(CommandBehavior behavior)

at Microsoft.Data.Sqlite.SqliteCommand.ExecuteNonQuery()

at Microsoft.Intune.Inventory.Service.Harvester.Database.SqliteHarvesterDatabase.SaveEventToDb(Event eventToCollect)

Configuration Manager Trace Log Tool - [C:\Program Files\Microsoft Device Inventory Agent\Logs\IntuneInventoryHarvesterLog.log]

File Tools Window Help

Log Text

8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_systemenclosure_serialnumber_model
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_systemenclosure_serialnumber_securitybreach
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_systemenclosure_serialnumber_serialnumber
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_systemenclosure_serialnumber_sku
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_systemenclosure_serialnumber_smbiosassettag
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_systemenclosure_serialnumber_status
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_systemenclosure_serialnumber_visaalarmequipped
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_tpm_singleton_manufacturer
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_tpm_singleton_manufacturerid
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_tpm_singleton_manufacturerversion
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_tpm_singleton_productname
8/25/2024 5:15:24 AM [Information] Processing PolicyId=a791f109-7a7b-46dc-8633-cd97bb4746b9 SettingId=windows_deviceinventory_tpm_singleton_specversion
8/25/2024 5:15:24 AM [Information] Completed harvesting policies.
8/25/2024 5:15:24 AM [Information] Beginning upload for 7 data sets.
8/25/2024 5:15:24 AM [Information] UpdateData: Successfully updated HarvesterEntitySet Cpu in sharedDB
8/25/2024 5:15:24 AM [Information] UpdateData: Successfully updated HarvesterEntitySet DiskDrive in sharedDB
8/25/2024 5:15:24 AM [Information] UpdateData: Successfully updated HarvesterEntitySet EncryptableVolume in sharedDB
8/25/2024 5:15:24 AM [Information] UpdateData: Successfully updated HarvesterEntitySet LogicalDrive in sharedDB
8/25/2024 5:15:24 AM [Information] UpdateData: Successfully updated HarvesterEntitySet OsVersion in sharedDB
8/25/2024 5:15:24 AM [Information] UpdateData: Successfully updated HarvesterEntitySet SystemEnclosure in sharedDB
8/25/2024 5:15:24 AM [Information] UpdateData: Successfully updated HarvesterEntitySet Tpm in sharedDB
8/25/2024 5:15:24 AM [Information] HeartbeatSync start for runid 3faf81ed-7df7-4b54-9849-17f8e94b77db.

Date/Time: Component:
Thread: Source:
8/24/2024 12:33:38 PM [Information] Initializing table creation

↑ C Program Files > Microsoft Device Inventory Agent >

Name	Date modified	Type	Size
JG-L			
IntuneInventoryHarvesterLog.log	8/25/2024 7:19 AM	Text Document	
InventoryAdaptor.log	8/25/2024 7:19 AM	Text Document	

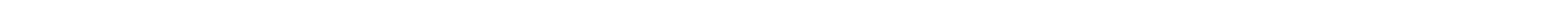
- If the agent is missing or the resource explorer is missing entities, check out the harvester log!!



The (not) Final overview

- We looked at Device Query
- We examined Device Inventory

Let's put it together!





The (not) Final overview

Adding Device Query first...





Are you ready?





Multi Device Query?

[In development - Microsoft Intune](#) | [Microsoft Learn](#)

Device Query for Multiple Devices

We're adding Device query for multiple devices. This feature allows you to gain **comprehensive insights** about your entire fleet of devices using Kusto Query Language (KQL) to query across collected **inventory data** for your devices.

Device query for multiple devices will be supported for devices running Windows 10 or later. This feature will be included as part of Advanced Analytics.

Applies to:



Please:
Take No Pictures



Please don't blog, post, or share any content publicly.





Device Query | Device Inventory

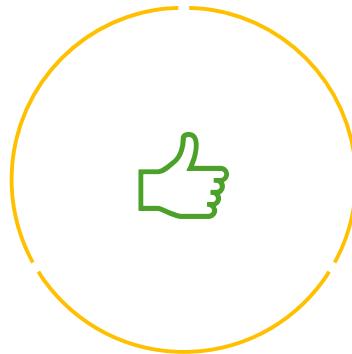
Device Query	Device Inventory	Multi Device Query
Troubleshooting	Reporting for single device	Reporting for all devices
Requires Additional Licensing	Intune Core Feature	Requires Additional Licensing
Live Data extracted from the device!	Queries the stored Inventory Data in Intune	Queries the stored Inventory Data in Intune
Uses the Intune Management Extension	Uses the Device Inventory Agent	Uses the data uploaded to the service by the Inventory Agent
Near Real time	Schedule	Real Time from the stored data

A wide-angle photograph of a large assembly of people in a port area. In the foreground, many individuals in military-style uniforms are seated in rows, facing a central stage or platform. The stage features several wooden tables and chairs. In the background, a large stone bridge arches over a body of water, with mountains visible under a cloudy sky.



Conclusion!

- Device Query is for troubleshooting
 - Device Query uses the IME and WNS
 - Device Inventory is for reporting
 - Device Inventory has its own agent
 - Device Inventory is for a single device
 - Device Inventory is now in Public Preview
 - Multi Device Query is for all devices
 - Multi Device Query is coming!
-



Thank You

