Intune Deep Dive:
How Policy And App Delivery Really works...

Sponsors

onedeploy
Windows Deployment Reimagined

ZERO TOUCH

Insight

control up

PATCH MY PC

robopack

numecent

Microsoft

eido

Rimo3

Rudy Ooms

Microsoft MVP 5 years
Patch My PC
Multiple Nicknames
-TroubleMaker
-The guy that reverse Code for fun
-Mister DLL
-Mister MDM

Hobbys? → The Above

*Joost Gelijsteen

Microsoft MVP 4 years
Secure at Work
Hobbys Sneaker, Sport
*It’s like toast… but with a J

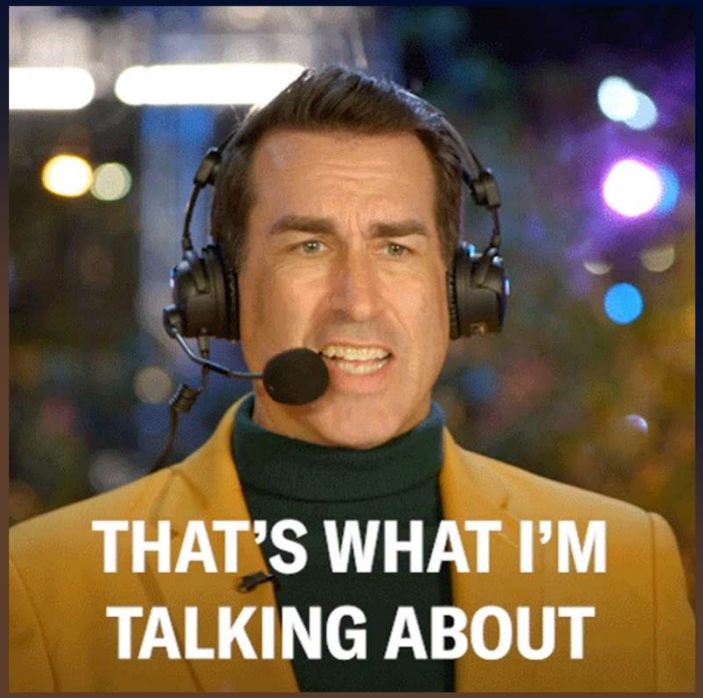
The Goal of this Webinar

Giving People
a clear understanding
How policies
coming down...

How it works...
How it doesn't Work!

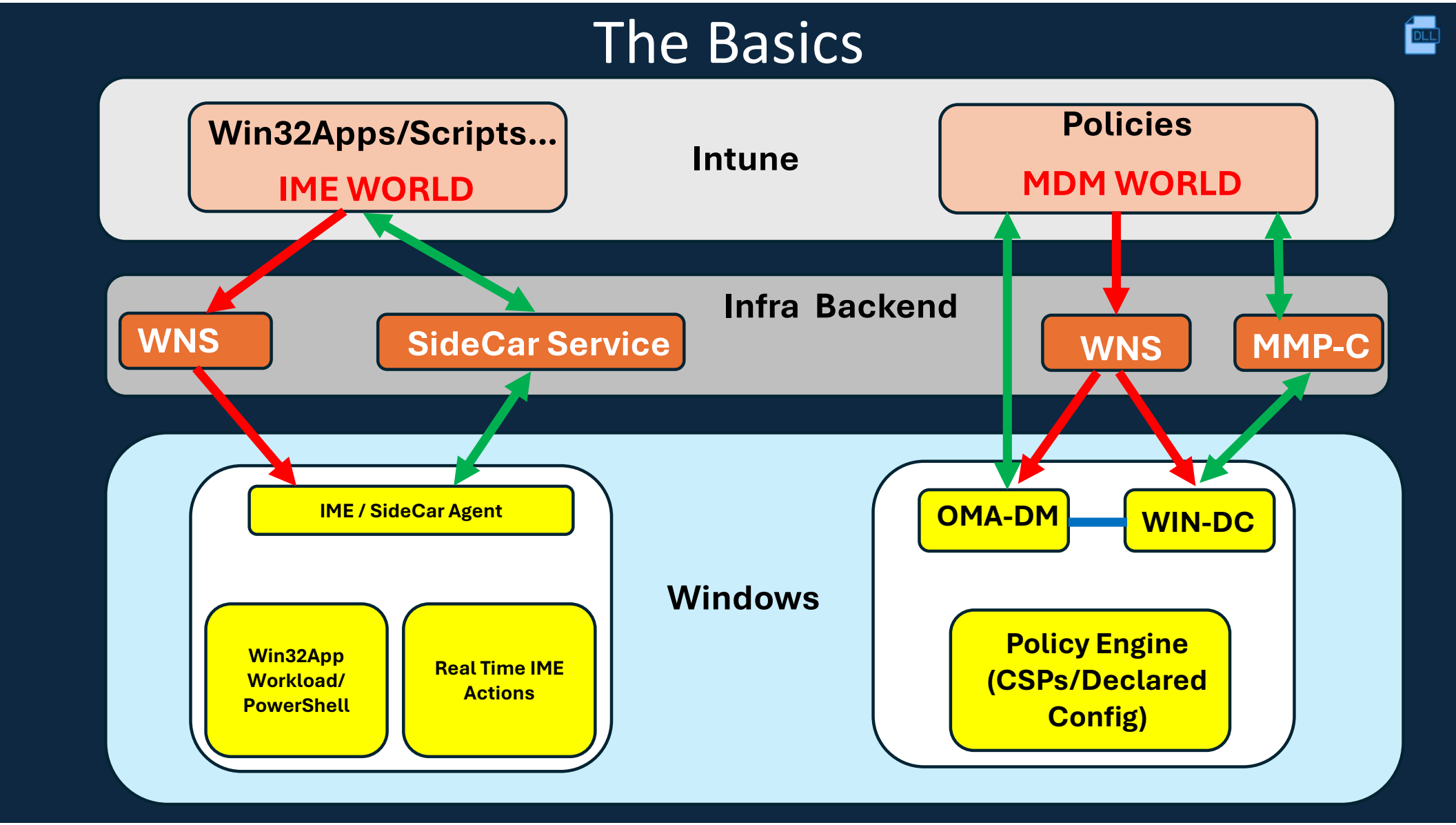
How Microsoft is
working on
Improvements

Agenda

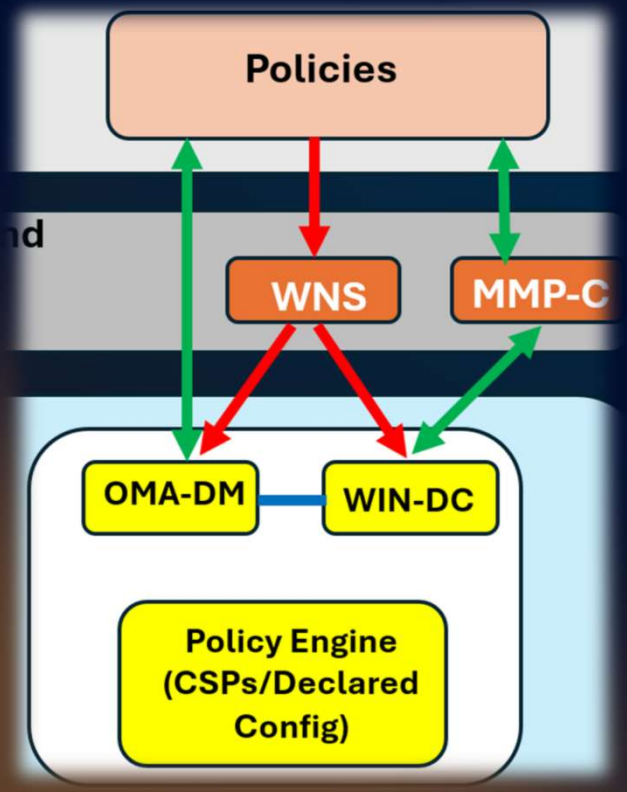


- The Basics
- The Full Policy Flow
- Windows Notification Service
- The Fastlane
  - Maybe More… ?(Depending on Time)

The Basics

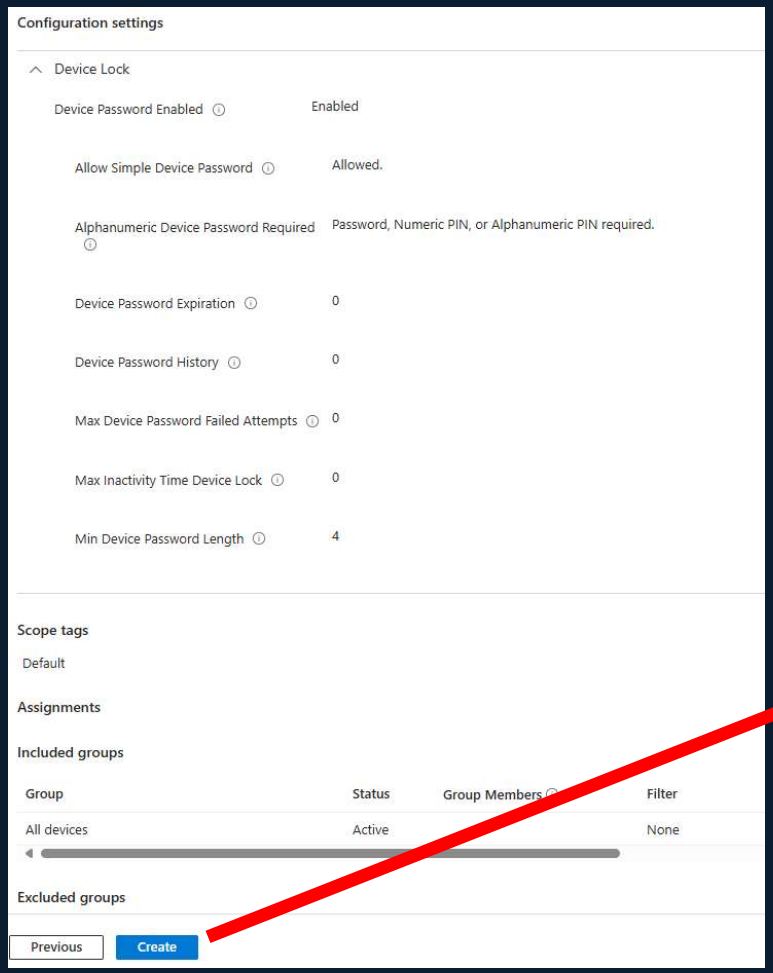

Win32Apps/Scripts...
IME WORLD
Intune
Policies
MDM WORLD
Infra Backend
WNS
SideCar Service
WNS
MMP-C
IME / SideCar Agent
Win32App Workload/ PowerShell
Real Time IME Actions
Windows
OMA-DM
WIN-DC
Policy Engine (CSPs/Declared Config)

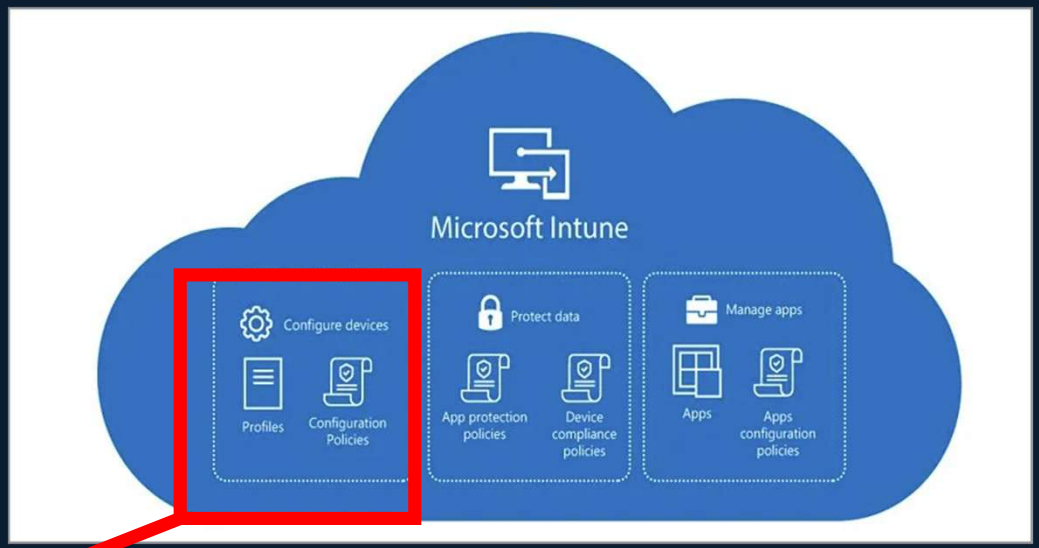
How Intune policies are delivered



Let's first examine how and when policies are sent over

Deploying Policies

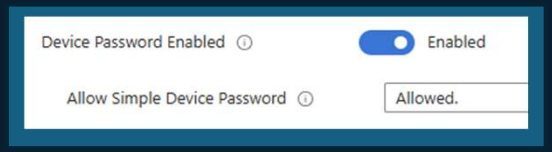


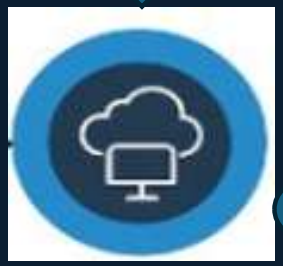


Let’s start creating a Policy

How Intune policies are delivered

1 Configure Policy in Intune





2 Intune sends policies in SyncML (XML format).

3 MDM Client (OMA-DM) parses the SyncML payload.

4 The CSP engine applies it locally. (get / set / get

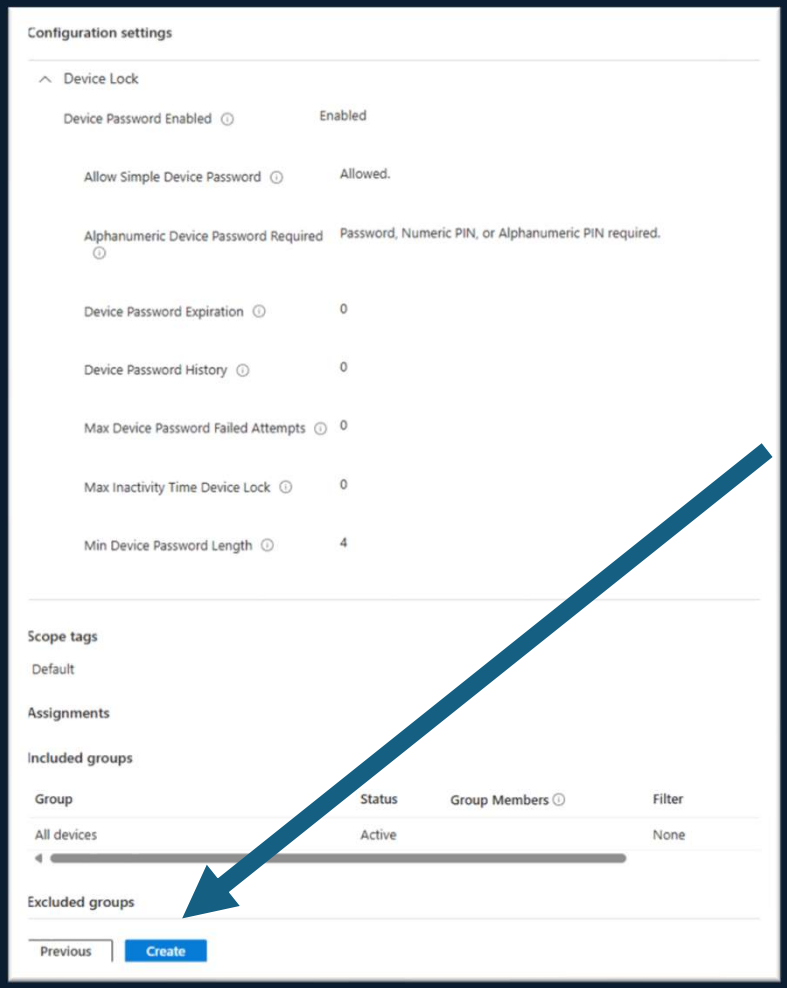
- ./Device/Vendor/MSFT/Policy
  - Config
    - {AreaName}
      - {PolicyName}
  - ConfigOperations
    - ADMXInstall
      - {AppName}
        - {SettingsType}
          - {AdmxFileId}
        - Properties
          - {SettingsType}
            - {AdmxFileId}
              - Version

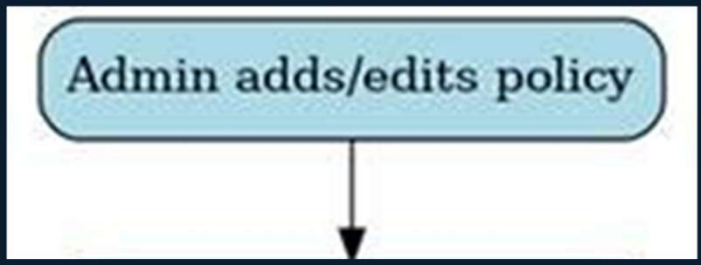
Let’s walk you through every step!

How Intune policies are delivered

How policies are sent over to the device ....in detail

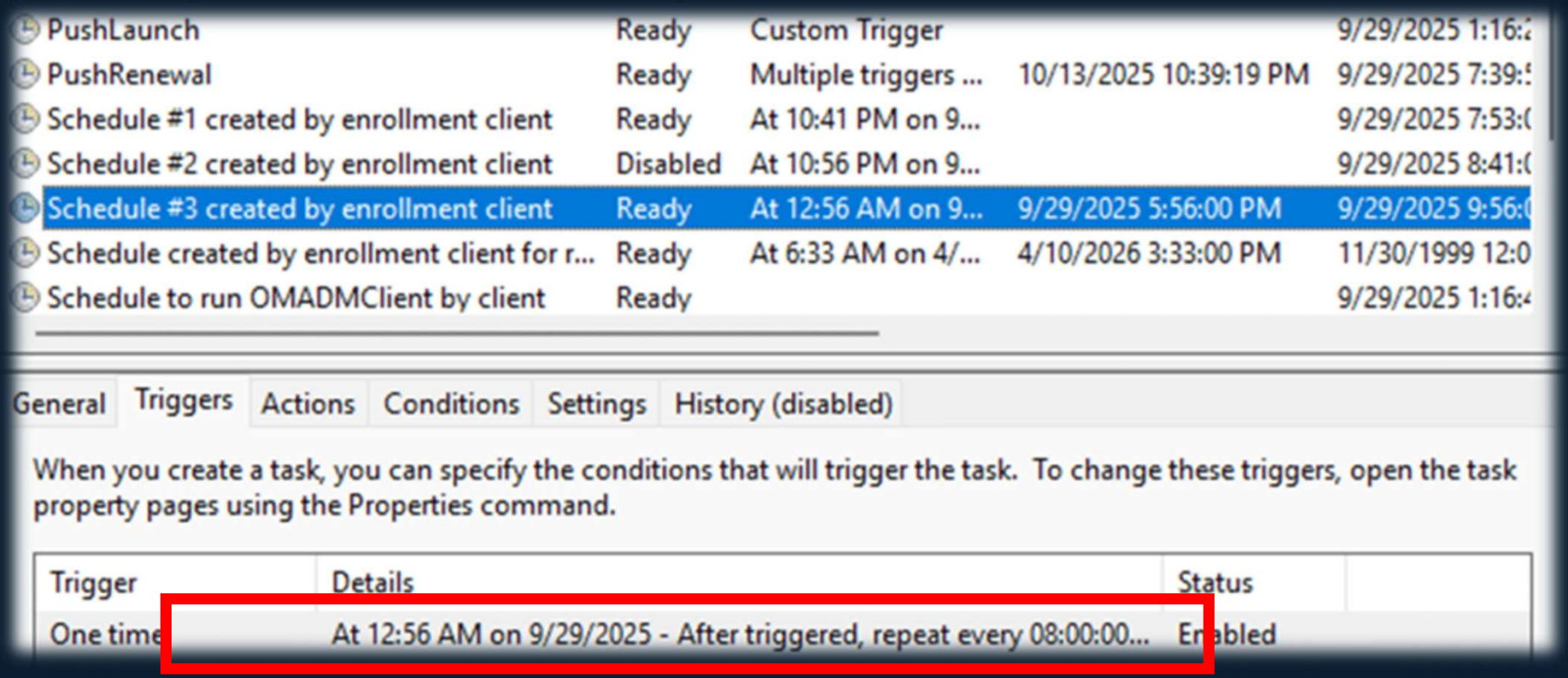
What happens when creating a new policy





So... again... lets create the policy first...

Do we need to wait?



Do we need to wait 8-hours before the policy arrives at the device?

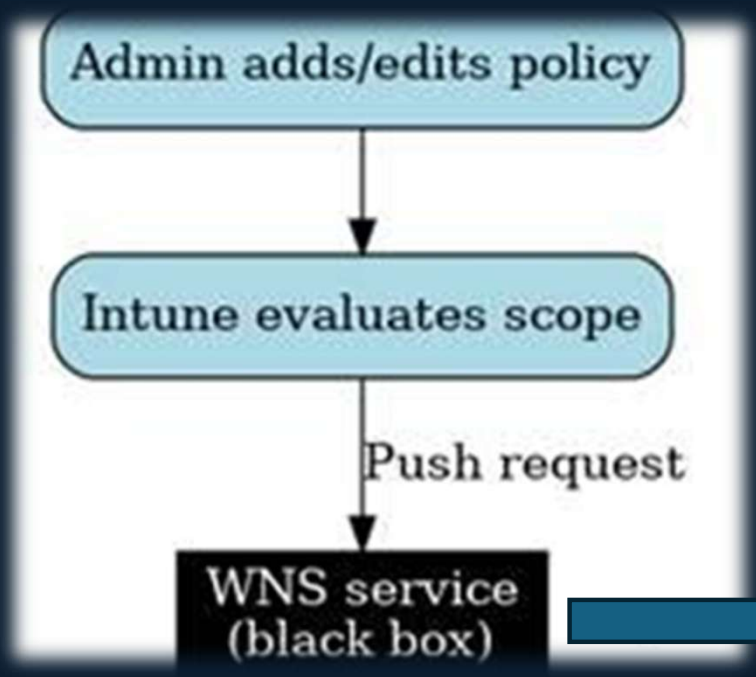
NOPE... let us explain!

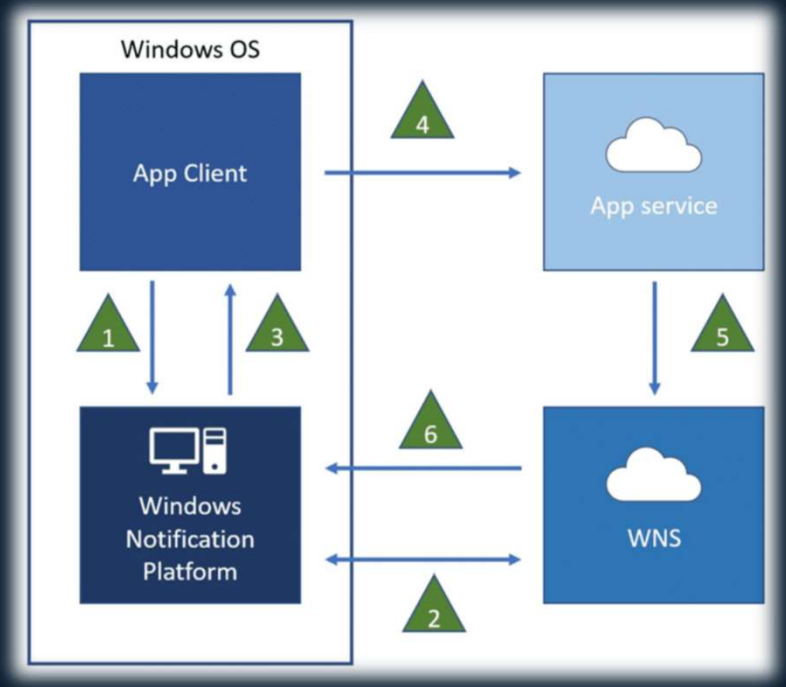
WNS

WNS Is
(still)
pretty
important
(for now)

- When we are changing targeting (adding or removing a device/user group)
- When we are Editing a payload (changing/adding a new Intune policy)
- When Entra group membership changes

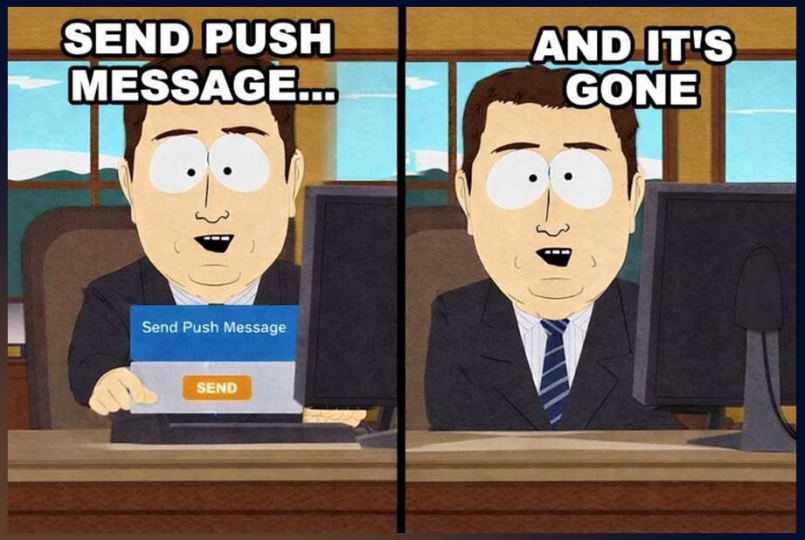
Intune will ask the WNS to deliver a message to the device, telling it to check in.





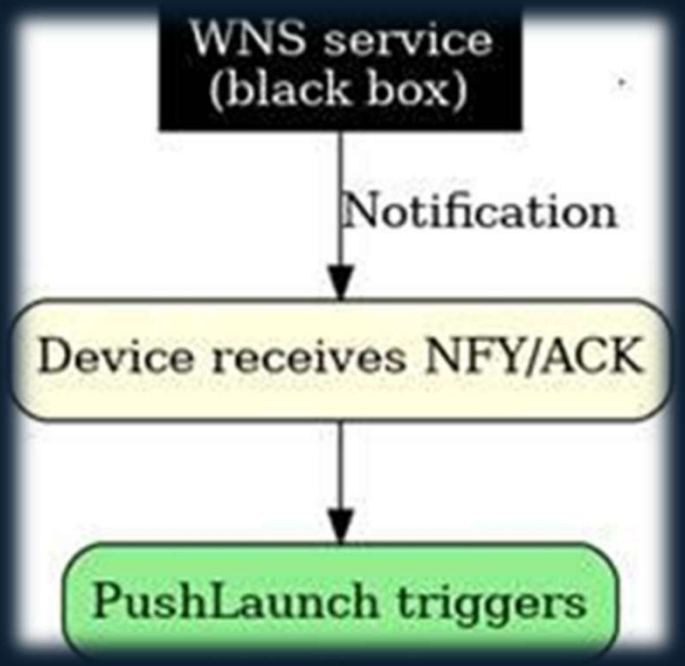
WNS the Blackbox

WNS “should” send a message to device asking it to check in..



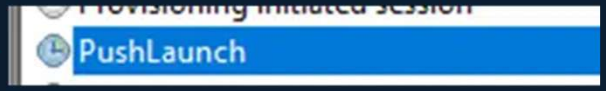
But with WNS being a black box.
That message might reach the device instantly, or it might disappear into the void without you ever knowing.

The Push Launch Task



The Push Message is just a wake-up call!

Once and if the push message arrives, It wakes up the PushLaunch scheduled task.



How does the PUSH wake up the PushLaunch Task?

The Push Launch Task

The PushLaunch task "listens" to a specific (75e0....)
Windows Notification Facility (WNF) State Change

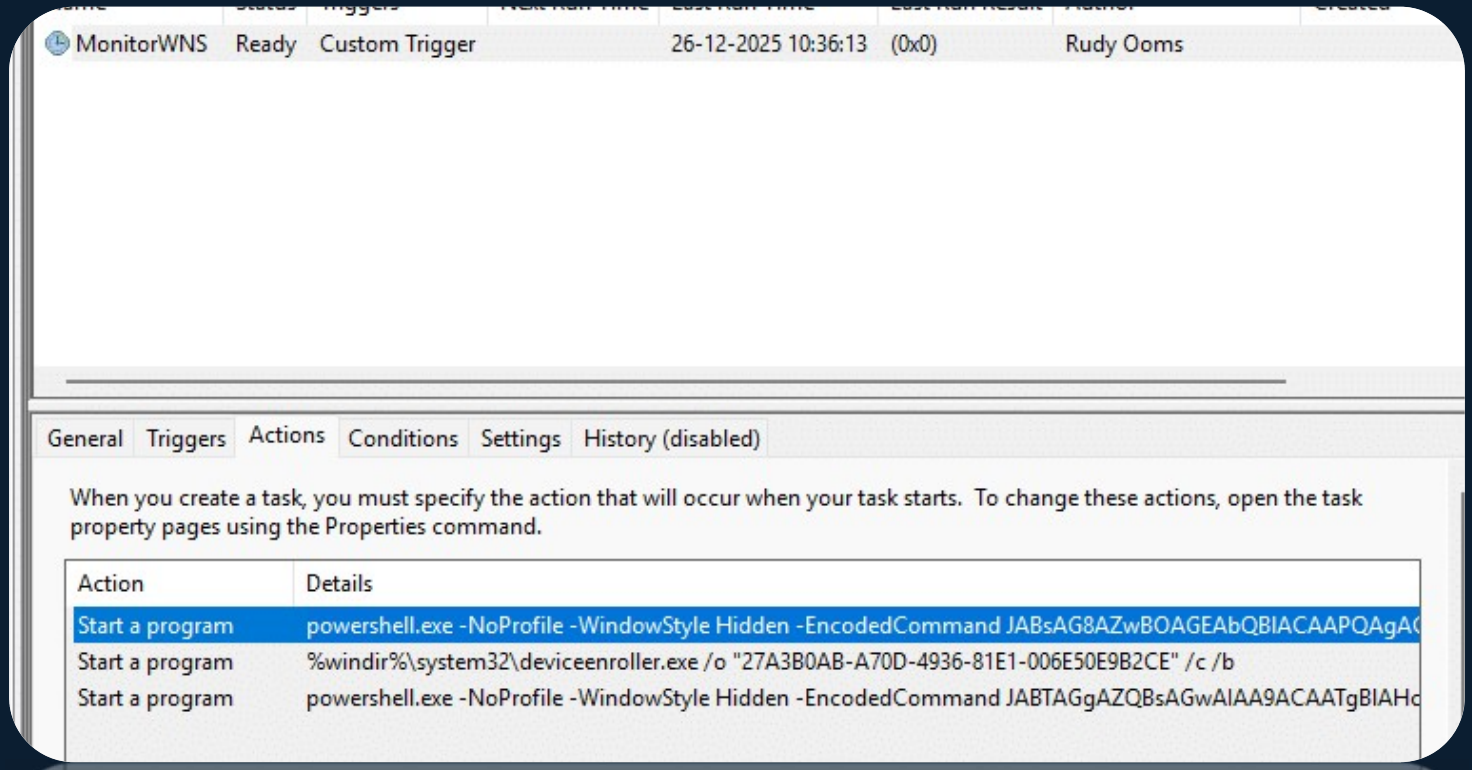
Something
Happened..
And now
we need to take action

```
    <UseUnifiedSchedulingEngine>true</UseUnifiedSchedulingEngine>
  </Settings>
  <Triggers>
    <WnfStateChangeTrigger>
      <StateName>75E0BEA328009213</StateName>
    </WnfStateChangeTrigger>
  </Triggers>
  <Actions Context="LocalSystem">
    <Exec>
      <Command>%windir%\system32\deviceenroller.exe</Command>
      <Arguments>/o "92C1BE86-70F0-4839-A677-5DA4D84FEE86" /c /z</Arguments>
    </Exec>
  </Actions>
```

```
232     75A0BCA328009213    WNF_ENTR_PUSH_RECEIVED  An enterprise WNS based push was received
```

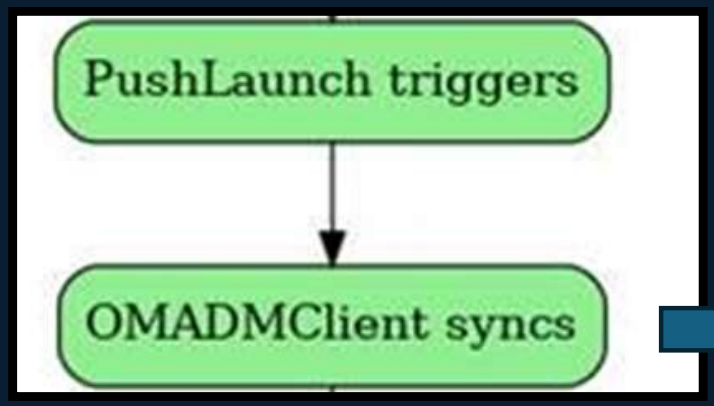
*WNF is the notification system within the Windows OS

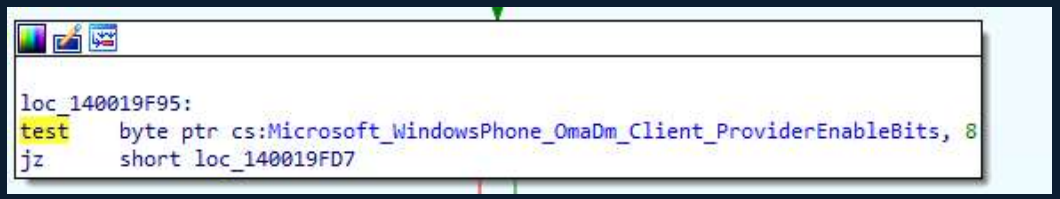
The Push Launch Task

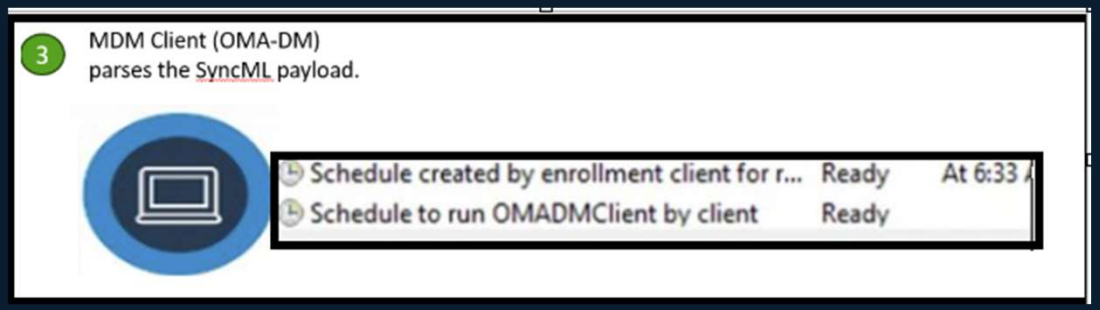


The funny thing is that you can create your own scheduled task with the same WNF State Change to trigger “something”
Don’t do that… Only Rudy Does that…

OMA-DM Client Sync



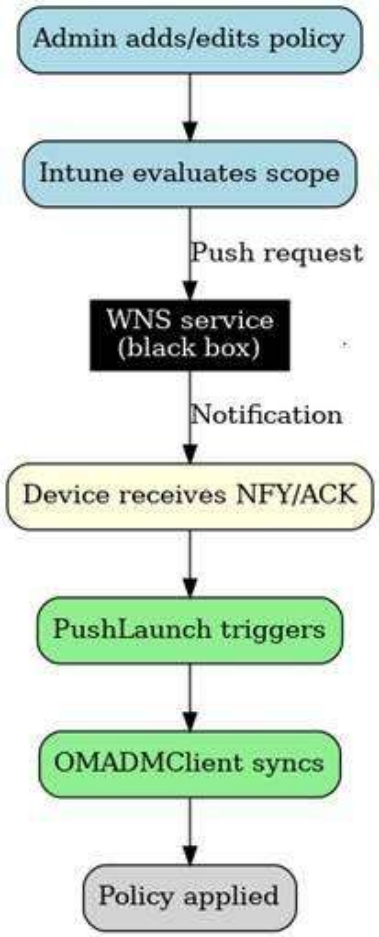




The PushLaunch task will *kick off The OMA-DM (phoneprovider) client and will reach out to Intune to fetch the policies

*with a delay of 5 minutes Queued Schedule
(if you don't have the feb 2026 update installed)

How A device receives a policy in nutshell

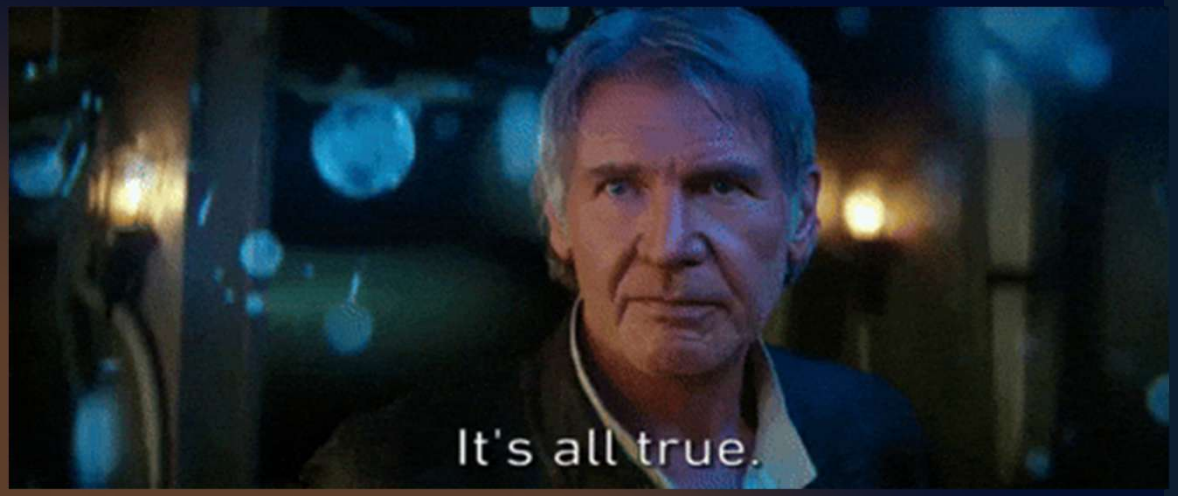


Summary:

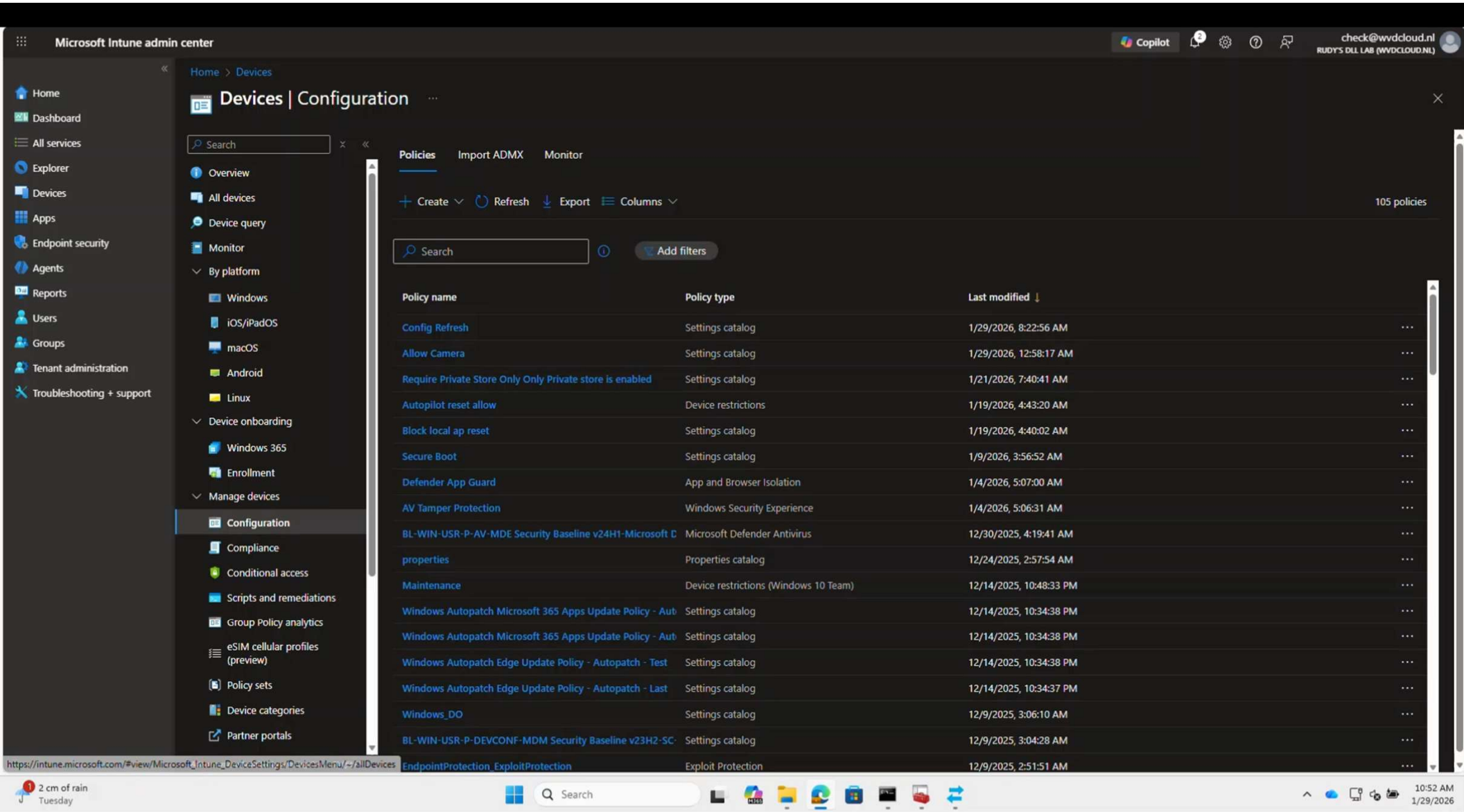
Editting/Adding a policy will ask WNS (the blackbox) to wake up the device.

It will trigger the pushlaunch, which will kick off the Sync with Intune to receive the policies.

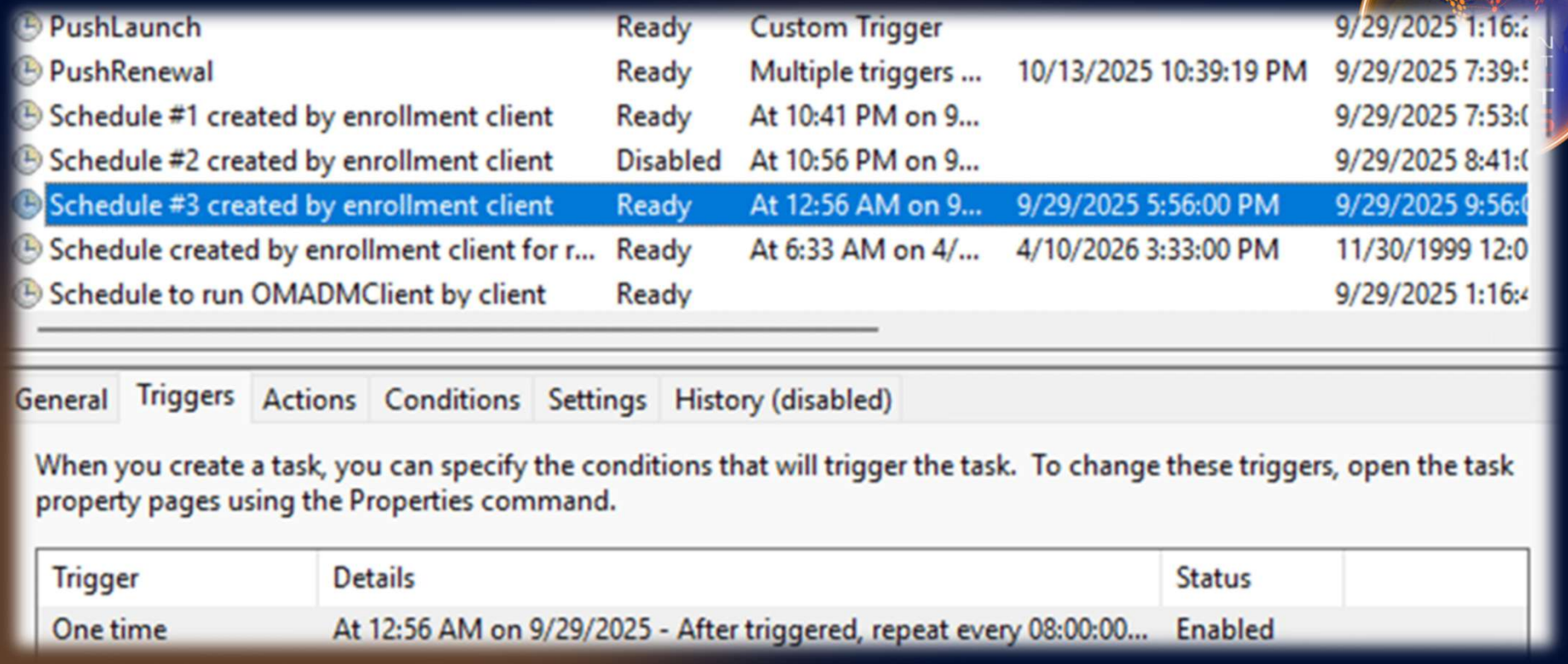
We are not lying?



Demo to
Prove it


Microsoft Intune admin center
Copilot
check@wvdcloud.nl
RUDY'S DLL LAB (WVDCLOUD.NL)
Home
Dashboard
All services
Explorer
Devices
Apps
Endpoint security
Agents
Reports
Users
Groups
Tenant administration
Troubleshooting + support
Home > Devices
Devices | Configuration
Search
Overview
All devices
Device query
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
Policies
Import ADMX
Monitor
Create
Refresh
Export
Columns
105 policies
Search
Add filters
Policy name
Policy type
Last modified
Config Refresh
Settings catalog
1/29/2026, 8:22:56 AM
Allow Camera
Settings catalog
1/29/2026, 12:58:17 AM
Require Private Store Only Only Private store is enabled
Settings catalog
1/21/2026, 7:40:41 AM
Autopilot reset allow
Device restrictions
1/19/2026, 4:43:20 AM
Block local ap reset
Settings catalog
1/19/2026, 4:40:02 AM
Secure Boot
Settings catalog
1/9/2026, 3:56:52 AM
Defender App Guard
App and Browser Isolation
1/4/2026, 5:07:00 AM
AV Tamper Protection
Windows Security Experience
1/4/2026, 5:06:31 AM
BL-WIN-USR-P-AV-MDE Security Baseline v24H1-Microsoft D
Microsoft Defender Antivirus
12/30/2025, 4:19:41 AM
properties
Properties catalog
12/24/2025, 2:57:54 AM
Maintenance
Device restrictions (Windows 10 Team)
12/14/2025, 10:48:33 PM
Windows Autopatch Microsoft 365 Apps Update Policy - Aut
Settings catalog
12/14/2025, 10:34:38 PM
Windows Autopatch Microsoft 365 Apps Update Policy - Aut
Settings catalog
12/14/2025, 10:34:38 PM
Windows Autopatch Edge Update Policy - Autopatch - Test
Settings catalog
12/14/2025, 10:34:38 PM
Windows Autopatch Edge Update Policy - Autopatch - Last
Settings catalog
12/14/2025, 10:34:37 PM
Windows_DO
Settings catalog
12/9/2025, 3:06:10 AM
BL-WIN-USR-P-DEVCONF-MDM Security Baseline v23H2-SC-
Settings catalog
12/9/2025, 3:04:28 AM
EndpointProtection_ExploitProtection
Exploit Protection
12/9/2025, 2:51:51 AM
https://intune.microsoft.com/#view/Microsoft_Intune_DeviceSettings/DevicesMenu/~/allDevices
2 cm of rain
Tuesday
Search
10:52 AM
1/29/2026

The 8 hour sync myth Busted



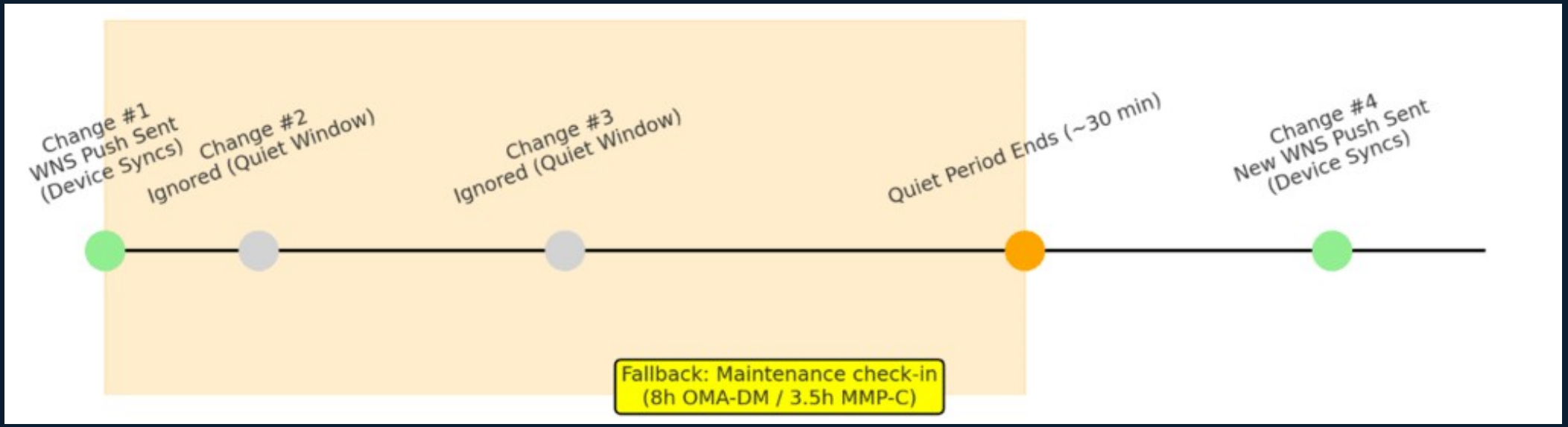
So we don't need to wait or manually sync our devices

(if the push.. Arrives..if.....)

Multiple Changes

What happens if we make multiple changes?

Be aware of throttling!! (for now)

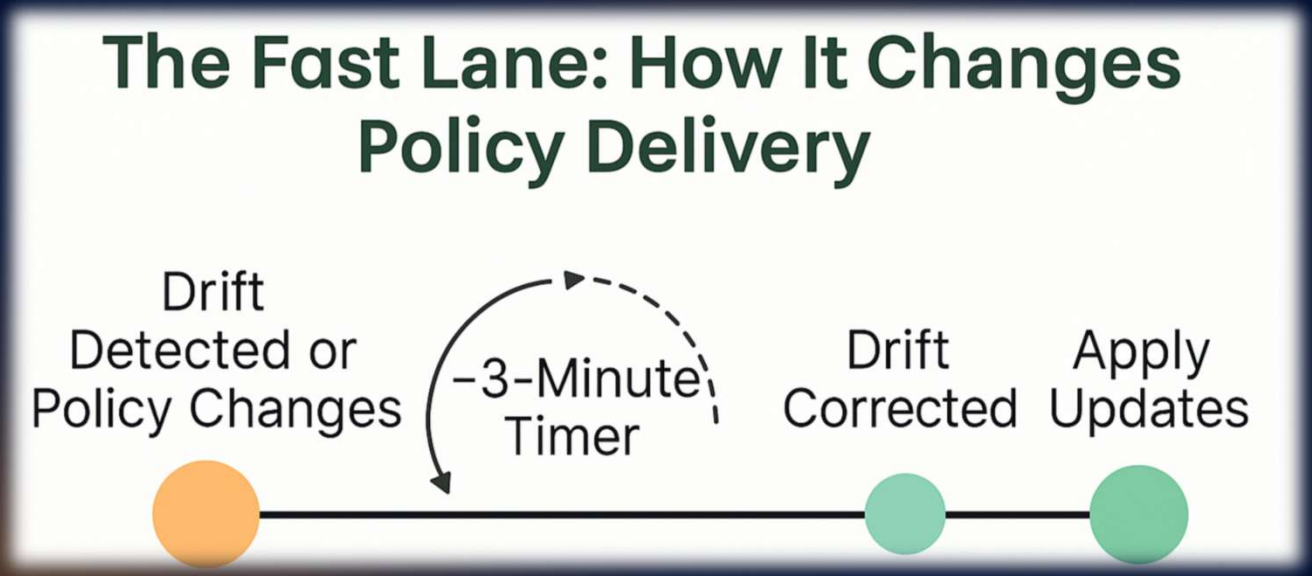


- Change #1 → WNS push almost immediate
- Change #2 and #3 (<30 min later) → bundled, resolved when device responds to the same notification
- Change #4 (>30 min later) → new WNS push delivered

It's not the 8-hour sync… it's the 30-minute throttle window that's the real delay.

That's where the new "Fast Lane" will improve things

The Fast Lane (when its there!)

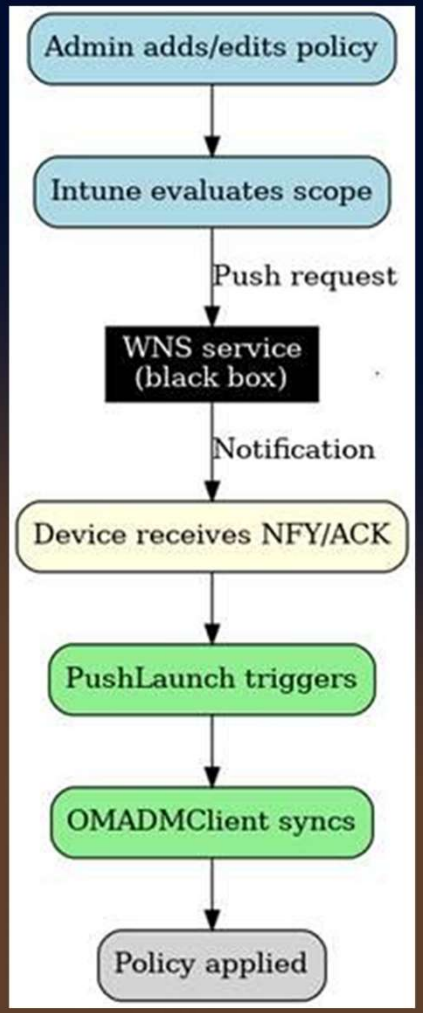


- When a drift (change) is detected, a timer starts.
- If no further changes happen within ~3 minutes, a notification is sent immediately.
- If more changes occur, the system queues them in small increments (up to ~10 minutes)
- So the 30 minutes Throttle will be gone!.

WNS did its job... what now?

How Policies Are being Applied

Applying the Policies on the device

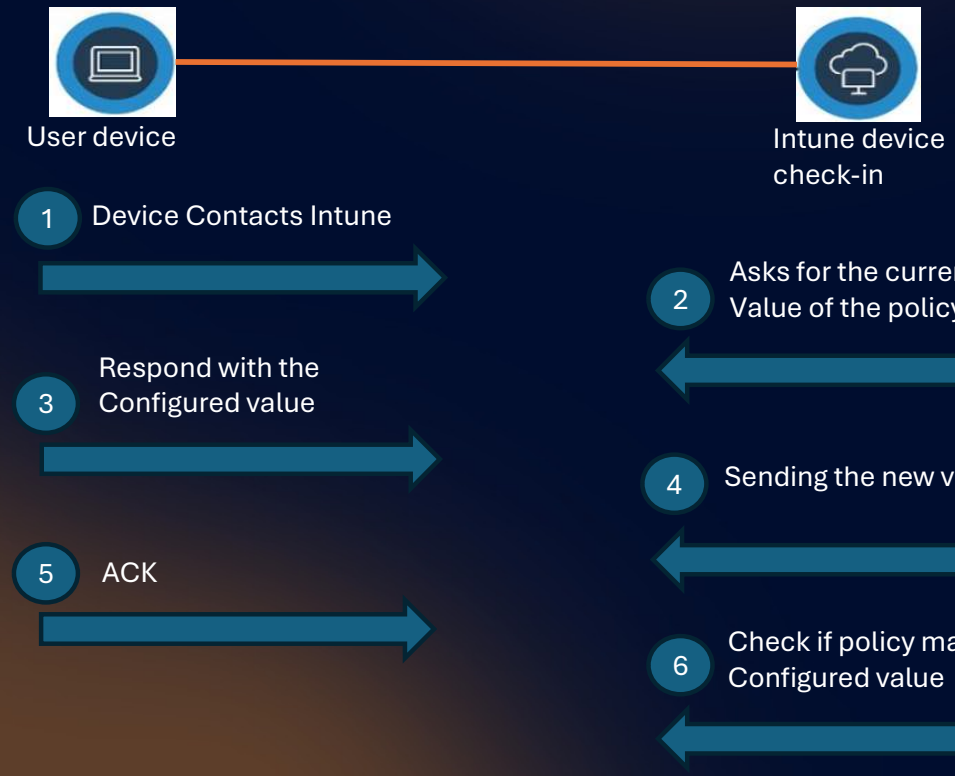


Now it's time for the Configuration Service Provider (CSP) to start applying the policies

4 The CSP engine applies it locally. (get / set / get

- ./Device/Vendor/MSFT/Policy
  - Config
    - {AreaName}
      - {PolicyName}
  - ConfigOperations
    - ADMXInstall
      - {AppName}
        - {SettingsType}
          - {AdmxFileId}
        - Properties
          - {SettingsType}
            - {AdmxFileId}
              - Version

Today's Enrollment & Policy Delivery



Policies delivered using OMA-DM Protocol (SyncML format)

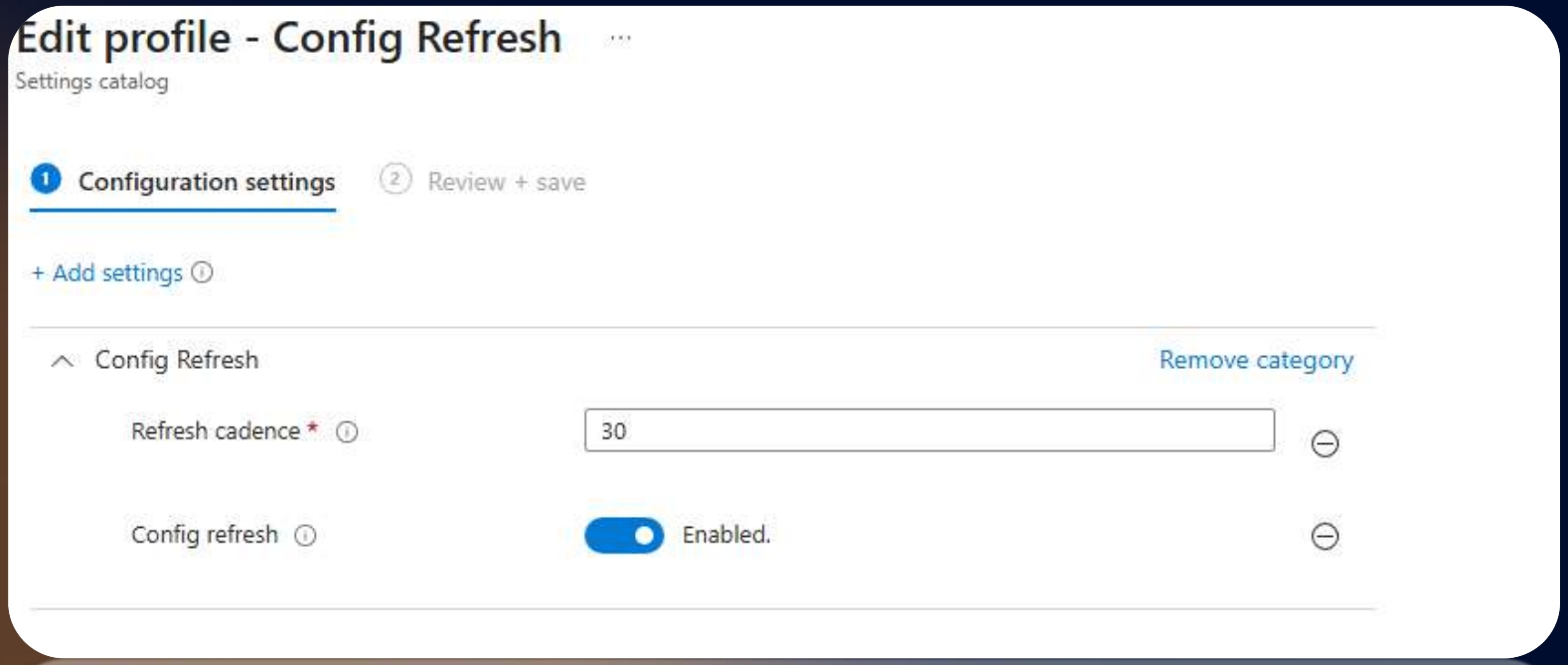
Multiple round trips PER policy: **get → set → get**

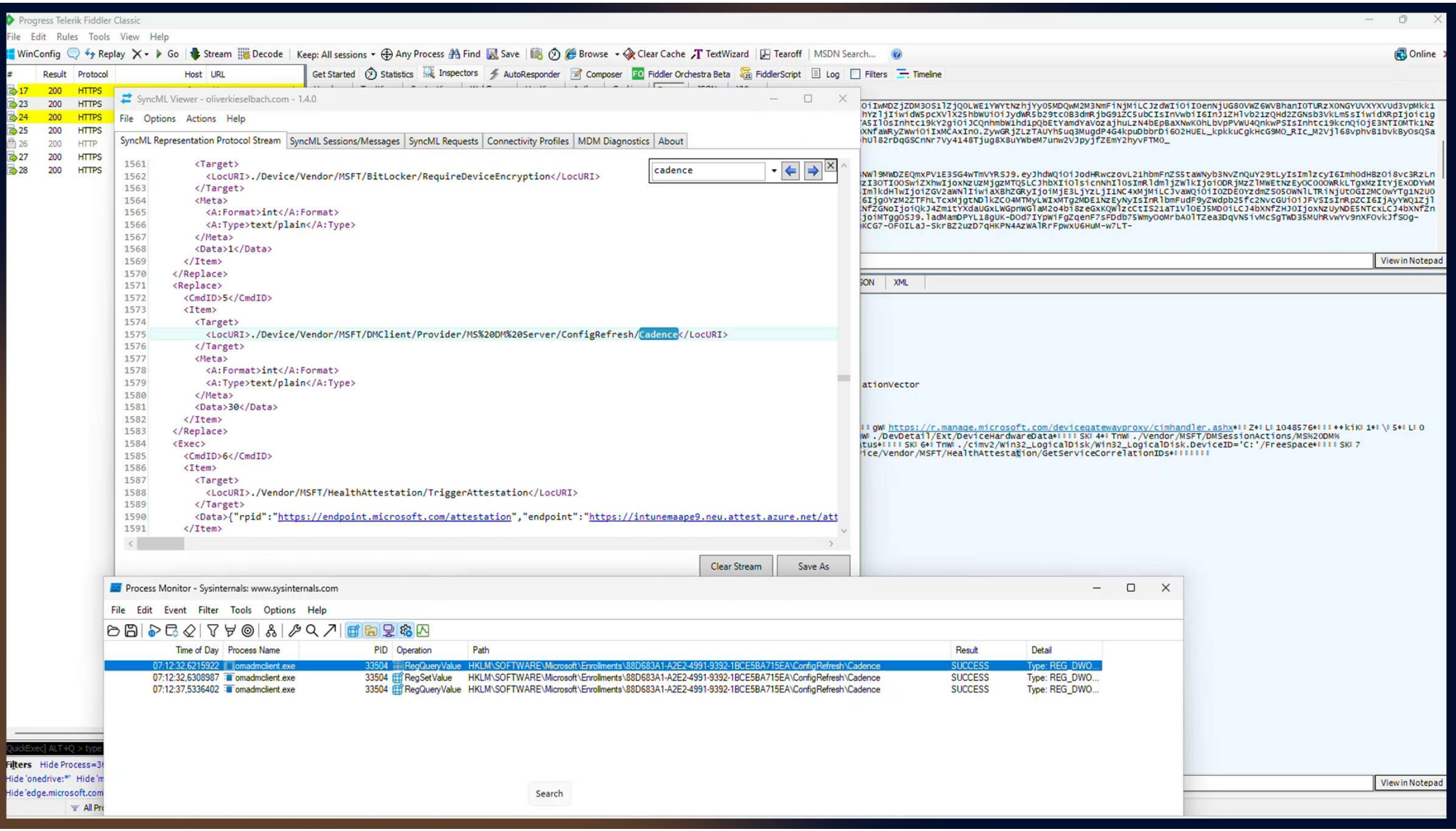
Today’s Enrollment & Policy Delivery

Example: Setting the Config Refresh Cadence to 30 Minutes

(it was 45)

Get 45 / Set 30 / Get 30




Progress Telerik Fiddler Classic
SyncML Viewer - oliverkieselbach.com - 1.4.0
SyncML Representation Protocol Stream
SyncML Sessions/Messages
SyncML Requests
Connectivity Profiles
MDM Diagnostics
About
cadence
<Target>
<LocURI>./Device/Vendor/MSFT/BitLocker/RequireDeviceEncryption</LocURI>
</Target>
<Meta>
<A:Format>int</A:Format>
<A:Type>text/plain</A:Type>
</Meta>
<Data>1</Data>
</Item>
</Replace>
<Replace>
<CmdID>5</CmdID>
<Item>
<Target>
<LocURI>./Device/Vendor/MSFT/DMClient/Provider/MS%20DM%20Server/ConfigRefresh/Cadence</LocURI>
</Target>
<Meta>
<A:Format>int</A:Format>
<A:Type>text/plain</A:Type>
</Meta>
<Data>30</Data>
</Item>
</Replace>
<Exec>
<CmdID>6</CmdID>
<Item>
<Target>
<LocURI>./Vendor/MSFT/HealthAttestation/TriggerAttestation</LocURI>
</Target>
<Data>{"rpid":"https://endpoint.microsoft.com/attestation","endpoint":"https://intunemaape9.neu.attest.azure.net/att
</Item>
Clear Stream
Save As
Process Monitor - Sysinternals: www.sysinternals.com
Time of Day
Process Name
PID
Operation
Path
Result
Detail
07:12:32,6215922 omadmclient.exe 33504 RegQueryValue HKLM\SOFTWARE\Microsoft\Enrollments\88D683A1-A2E2-4991-9392-1BCE5BA715EA\ConfigRefresh\Cadence SUCCESS Type: REG_DWO...
07:12:32,6308987 omadmclient.exe 33504 RegSetValue HKLM\SOFTWARE\Microsoft\Enrollments\88D683A1-A2E2-4991-9392-1BCE5BA715EA\ConfigRefresh\Cadence SUCCESS Type: REG_DWO...
07:12:37,5336402 omadmclient.exe 33504 RegQueryValue HKLM\SOFTWARE\Microsoft\Enrollments\88D683A1-A2E2-4991-9392-1BCE5BA715EA\ConfigRefresh\Cadence SUCCESS Type: REG_DWO...
Search

Latency

Why
Latency is
NOT funny
(well maybe a bit)

The Latency Bottleneck

- **Each policy requires multiple roundtrips (GET → SET → GET)**
  → Slows down deployment and increases failures
- **Throttling kicks in after too many requests**

(we have all been there… clicking the sync button a lot of times to speed it up)

→ Devices may stop processing policies until throttle is gone

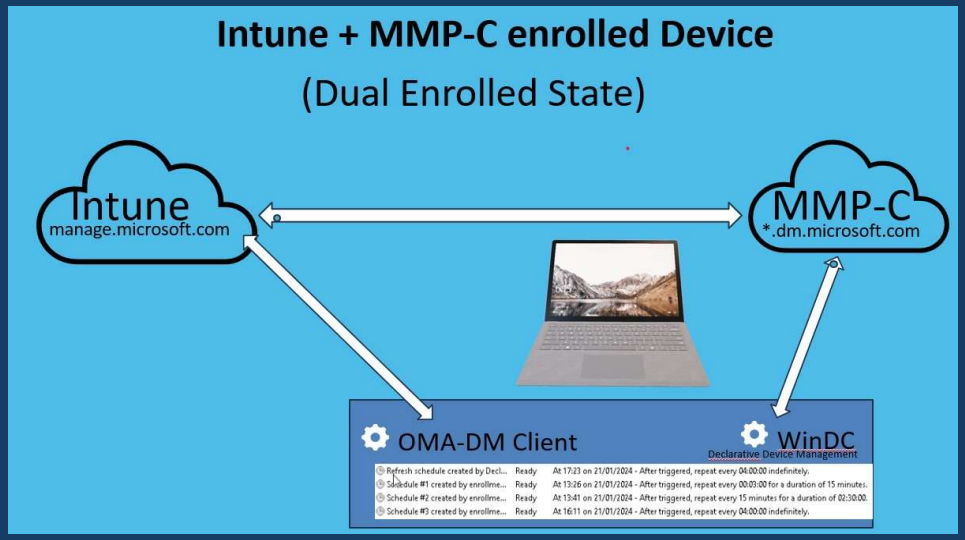
- **No offline enforcement**
  → Devices can't correct or apply policies unless connected to the cloud.
- **Admins get limited visibility when something fails**
  → Troubleshooting becomes reactive and time-consuming.

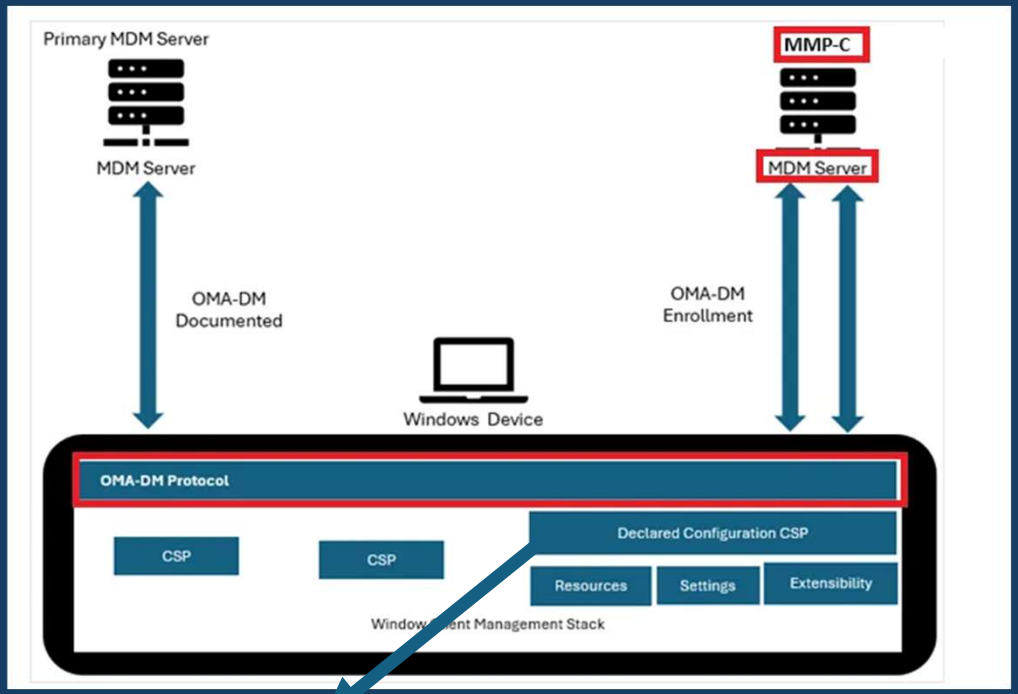
Yes (msft) can!

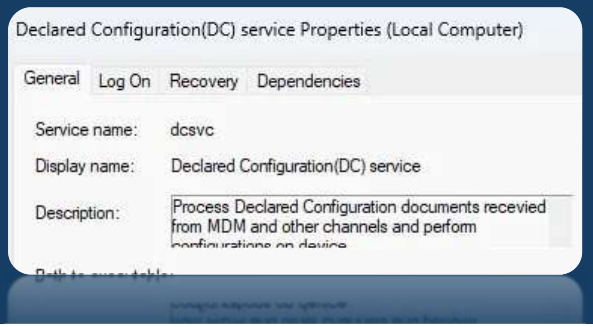
Fixing the Policy Latency

This is called a Linked/Dual Enrolled

All Devices now have both OMA-DM and WinDC channel



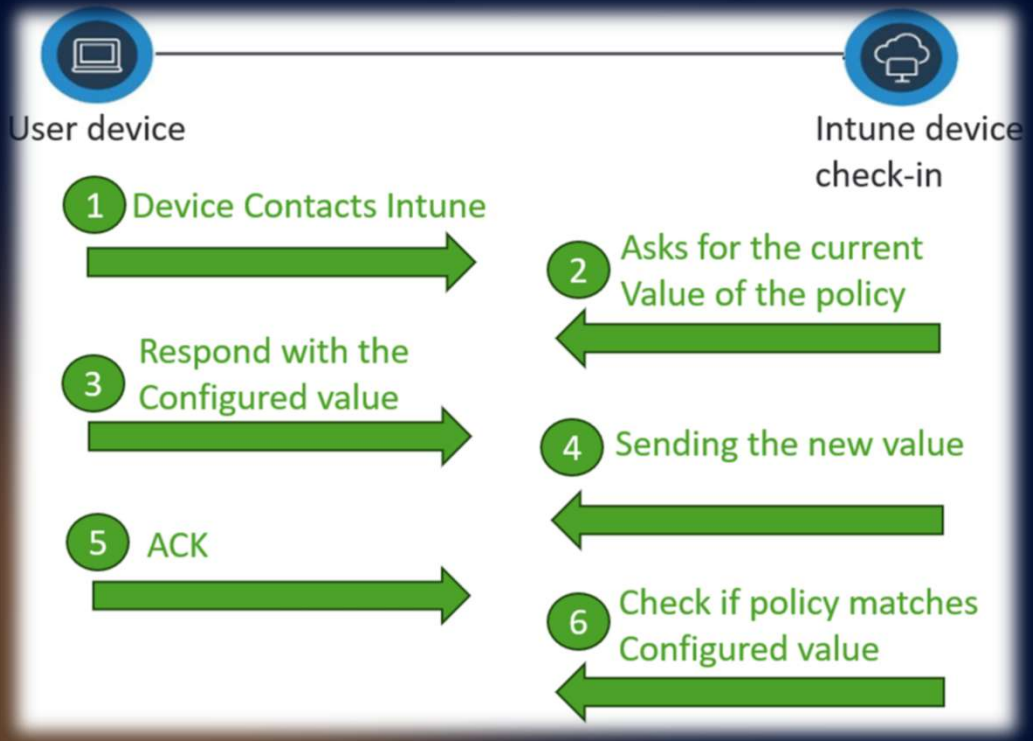




Which lets devices manage their own settings and send status updates to the management server on their own, without having to constantly check in

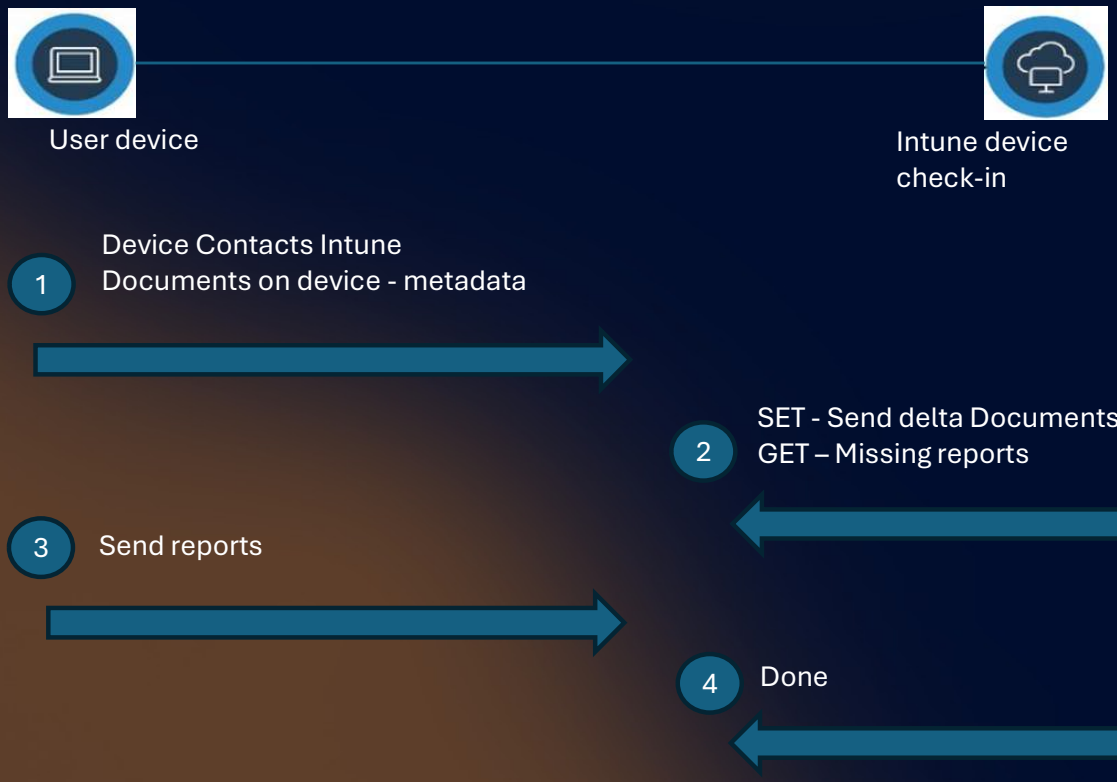
Declared Configuration Policies

Small Recap: The OLD Way
From Get – Set – Get….for every policy (slow…)



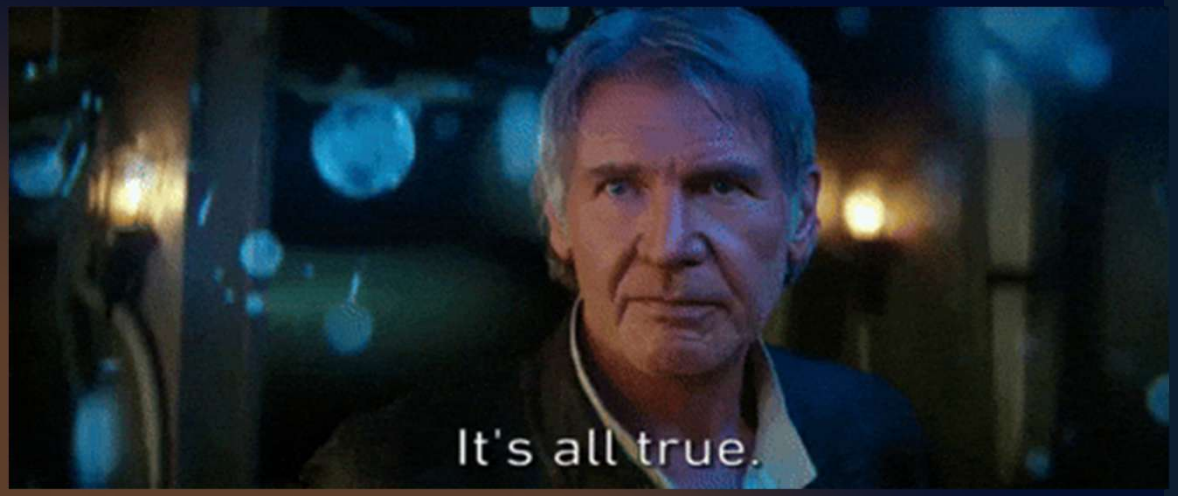
Declared Policies

To SET / GET…for everything at once (The Fastlane!)

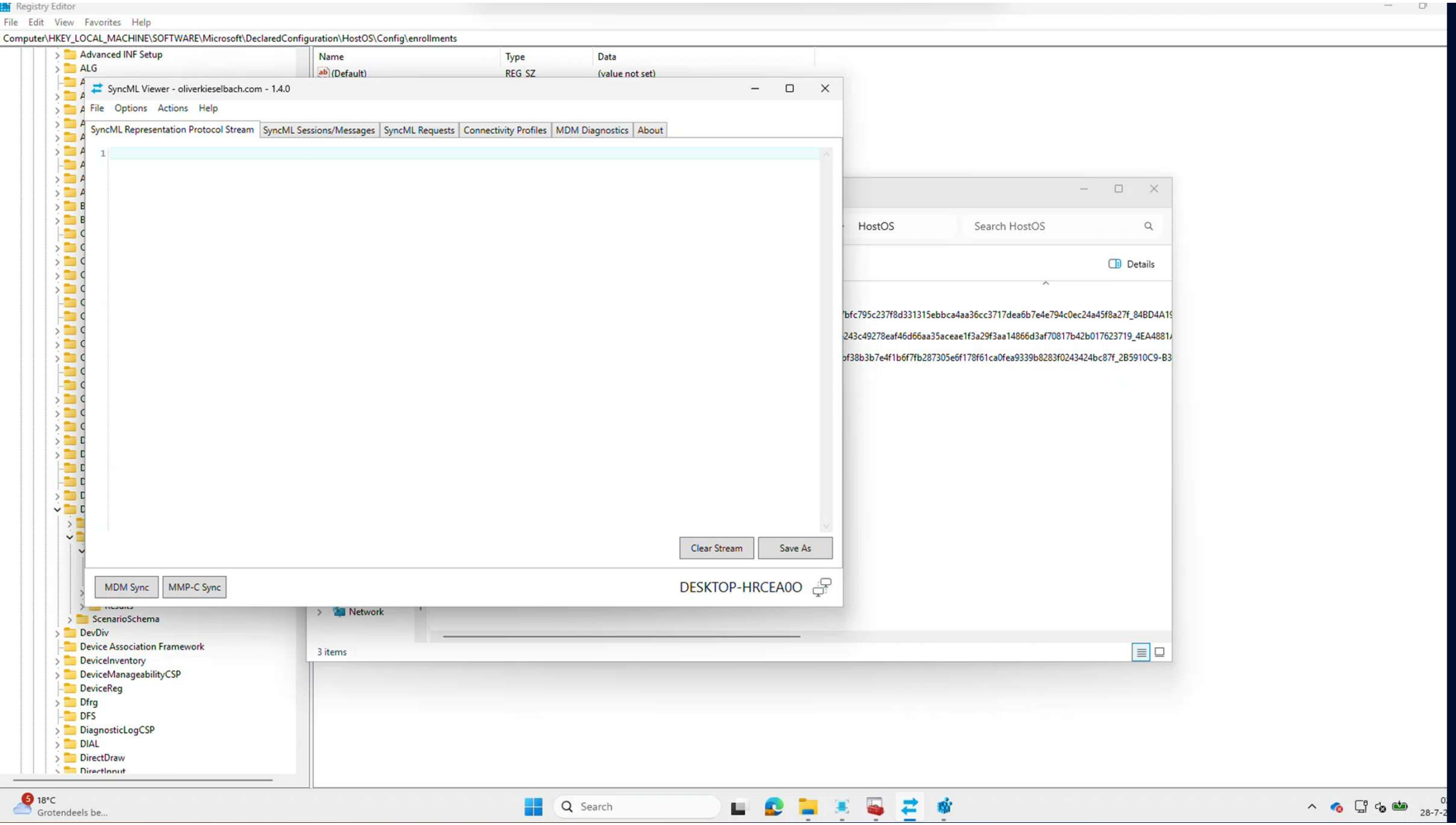


- batched declared configuration document that describes the entire desired state.
- Device will keep itself in a desired state Even offline!
- Will proactively checkin to intune if a change occurs

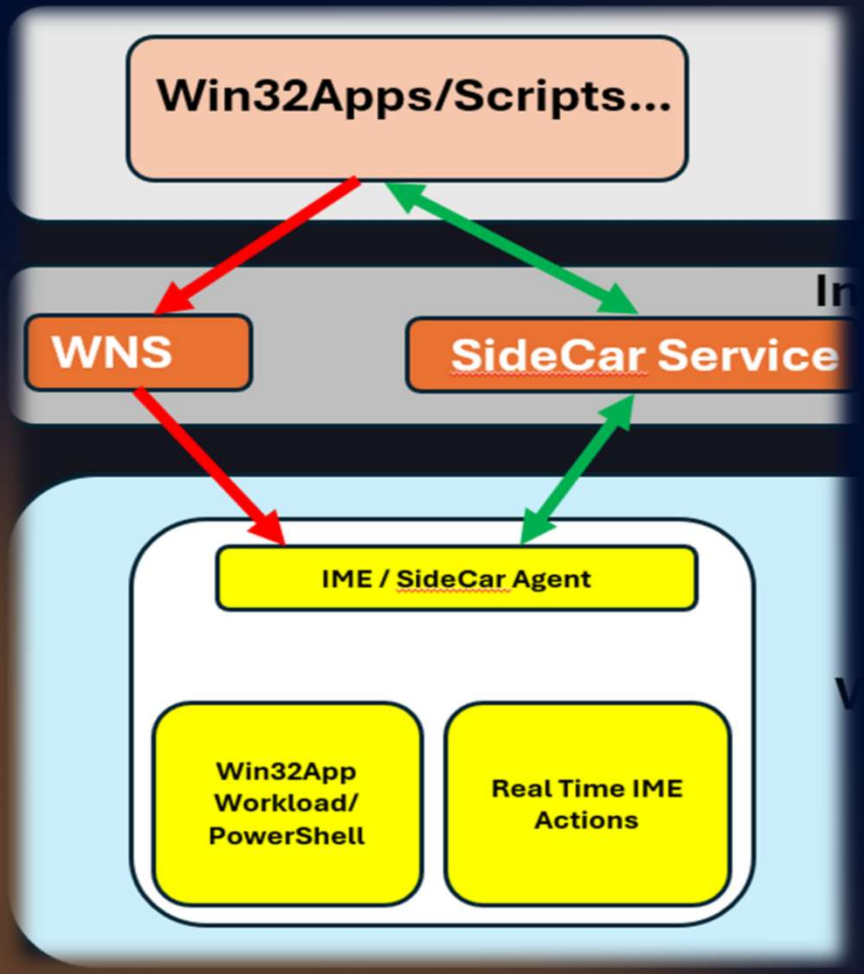
It’s Fast…. We Are not Lying



Demo to
Prove it


Registry Editor
File Edit View Favorites Help
Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\DeclaredConfiguration\HostOS\Config\enrollments
Name Type Data
(Default) REG_SZ (value not set)
SyncML Viewer - oliverkieselbach.com - 1.4.0
File Options Actions Help
SyncML Representation Protocol Stream
SyncML Sessions/Messages
SyncML Requests
Connectivity Profiles
MDM Diagnostics
About
Clear Stream
Save As
MDM Sync
MMP-C Sync
DESKTOP-HRCEA0O
HostOS
Search HostOS
Details
3 items
Network
ScenarioSchema
DevDiv
Device Association Framework
DeviceInventory
DeviceManageabilityCSP
DeviceReg
Dfrg
DFS
DiagnosticLogCSP
DIAL
DirectDraw
18°C
Search

What about the Apps/Scripts?



How the IME / SideCar agent is also evolving

What about Apps?

```
public CheckinIntervalManager(ITimer timerWrapper)
{
  Guard.ArgumentNotNull((object) timerWrapper, nameof (timerWrapper));
  this.timerWrapper = timerWrapper;
  this.shouldUpdateTimer = false;
  this.intervalForTimer = 3600000.0;
  this.setTimerIntervalToDefaultFlag = false;
}

public void SetTimerInterval(int? nextCheckinIntervalnMins)
{
  this.shouldUpdateTimer = true;
  this.intervalForTimer = !nextCheckinIntervalnMins.HasValue ? 3600000.0 : (double) (nextCheckinIntervalnMins.Value
  AppWorkloadLog.TraceInformation($"[Win32App][CheckinIntervalManager] SetTimerInterval. {$"{"shouldUpdateTimer"}:
}

public void SetTimerIntervalToDefault()
{
  this.setTimerIntervalToDefaultFlag = true;
  AppWorkloadLog.TraceInformation("[Win32App][CheckinIntervalManager] SetTimerIntervalToDefault.");
}

public void UpdateTimerIntervalForNextCheckin()
{
  AppWorkloadLog.TraceInformation($"[Win32App][CheckinIntervalManager] UpdateTimerIntervalForNextCh
  if (this.timerWrapper.GetInterval() == this.intervalForTimer)
    return;
  AppWorkloadLog.TraceInformation($"[Win32App][CheckinIntervalManager] {nameof (UpdateTimerInterval
  if (this.setTimerIntervalToDefaultFlag)
    this.timerWrapper.UpdateInterval(3600000.0);
  else if (this.shouldUpdateTimer)
    this.timerWrapper.UpdateInterval(this.intervalForTimer);
  AppWorkloadLog.TraceInformation($"[Win32App][CheckinIntervalManager] {nameof (UpdateTimerInterval
```

Today, required Apps and PowerShell Scripts Rely on PULL instead of Push

```
if (emsPolicyList1 != null && emsPolicyList1.Any<EmsPolicy>() && AADJoinHelper.IsDevi
{
  Log.TraceInformation("[PowerShell] Calculating effectivePolicies.");
  foreach (EmsPolicy emsPolicy in emsPolicyList1)
  {
    if (emsPolicy.ExecutionContext != 1)
      emsPolicyList2.Add(emsPolicy);
    else
      Log.TraceInformation("[PowerShell] This is not AADJ/HAADJ device, skip user cor
  }
  emsPolicyList1 = emsPolicyList2;
}
if (!flag1 && emsPolicyList1 != null && emsPolicyList1.Count > 0 && this.TimerInterna
{
  this.TimerInternal.Interval = 28800000.0;
  Log.TraceInformation($"[PowerShell] PS Script found, Timer is changed to {TimeSpan.
}
Log.TraceInformation("[PowerShell] After filter, get {0} policies for user {1} in ses
dictionary.Clear();
dictionary.Add("Step", (object) "PolicyProcess");
this.msg = string.Format((IFormatProvider) CultureInfo.InvariantCulture, "[PowerShell
dictionary.Add("EventMessage", (object) this.msg);
if (emsPolicyList1.Any<EmsPolicy>((Func<EmsPolicy, bool>) (policy => policy.PolicyTyp
```

PowerShell = 8 Hour
Required Apps = 1 hour

Asking the User to reboot

(logoff/logon → Session Change)


```
protected override void OnSessionChange(SessionChangeDescription changeDescription)
{
  Log.TraceWarning("Session change detected.");
  Microsoft.Management.Services.IntuneWindowsAgent.AgentCommon.TelemetryManager.LogInfo(nameof (OnSessionChange), nameof (OnSessionCha
  switch (changeDescription.Reason)
  {
    case SessionChangeReason.SessionLogon:
      Log.TraceWarning("OnSessionChange: Logon");
      Thread.Sleep(30000);
      Log.TraceWarning("[PowerShell] Request PS Scripts for user logon.");
      Log.TraceWarning("[PowerShell] Check flighting to call OnTimer from OnSessionChange.");
      this.CallSelectedOnTimer((object) new CheckinReasonTypes(CheckinReason.UserLogOn), (ElapsedEventArgs) null);
      this.ProcessAppsOnSessionChange();
      Log.TraceWarning("[RebootHelper] CheckRebootFlagAndPop start...");
      RebootHelper.CheckRebootFlagAndPop(RebootCheckCaller.SessionLogon);
      Log.TraceWarning("[RebootHelper] CheckRebootFlagAndPop complete");
      HSLog.TraceWarning("[HS] Request health script check in for user logon.");
      PolicyPoller.Instance.Schedule((object) new CheckinReasonTypes(CheckinReason.UserLogOn), (ElapsedEventArgs) null);
      break;
```

It works……

It will trigger the IME to kickoff the Win32 App and PowerShell Workload

Can MSFT make it better?

Fixing the Apps/Scripts

Latency

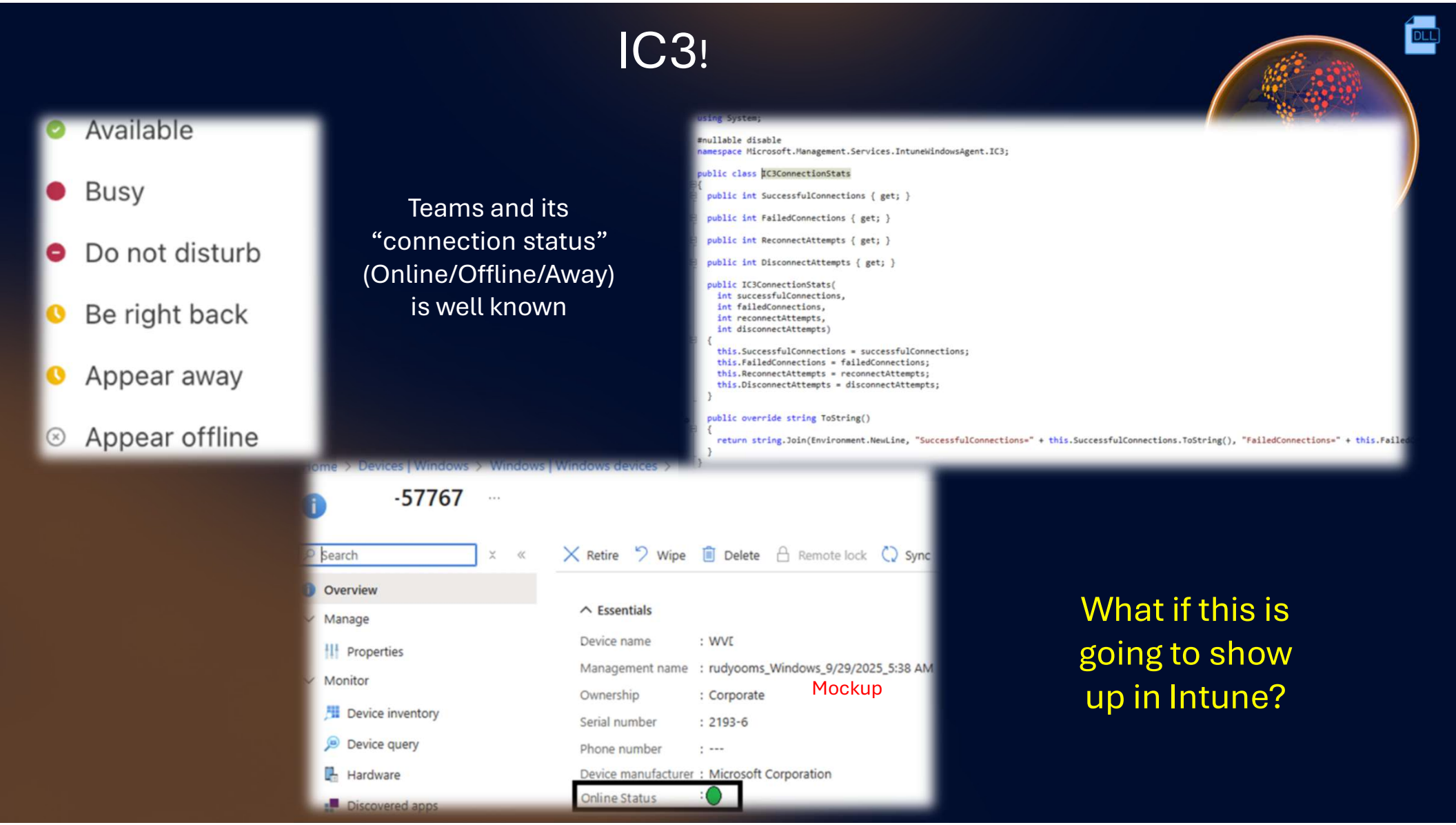
IC3!

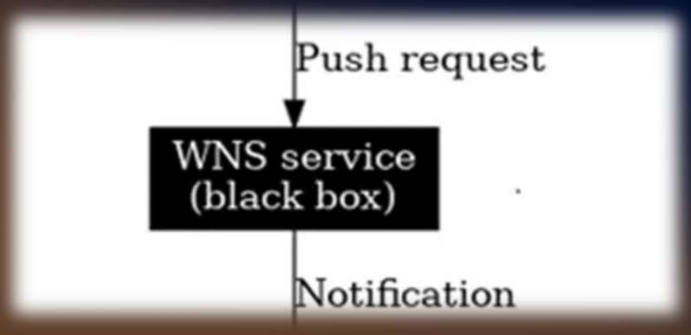
This is where IC3 Comes in

Intelligent Conversation and Communications Cloud (I 3xC)

IC3 is the core infrastructure platform powering Microsoft Teams and Azure Communication Services (ACS) for instant communication

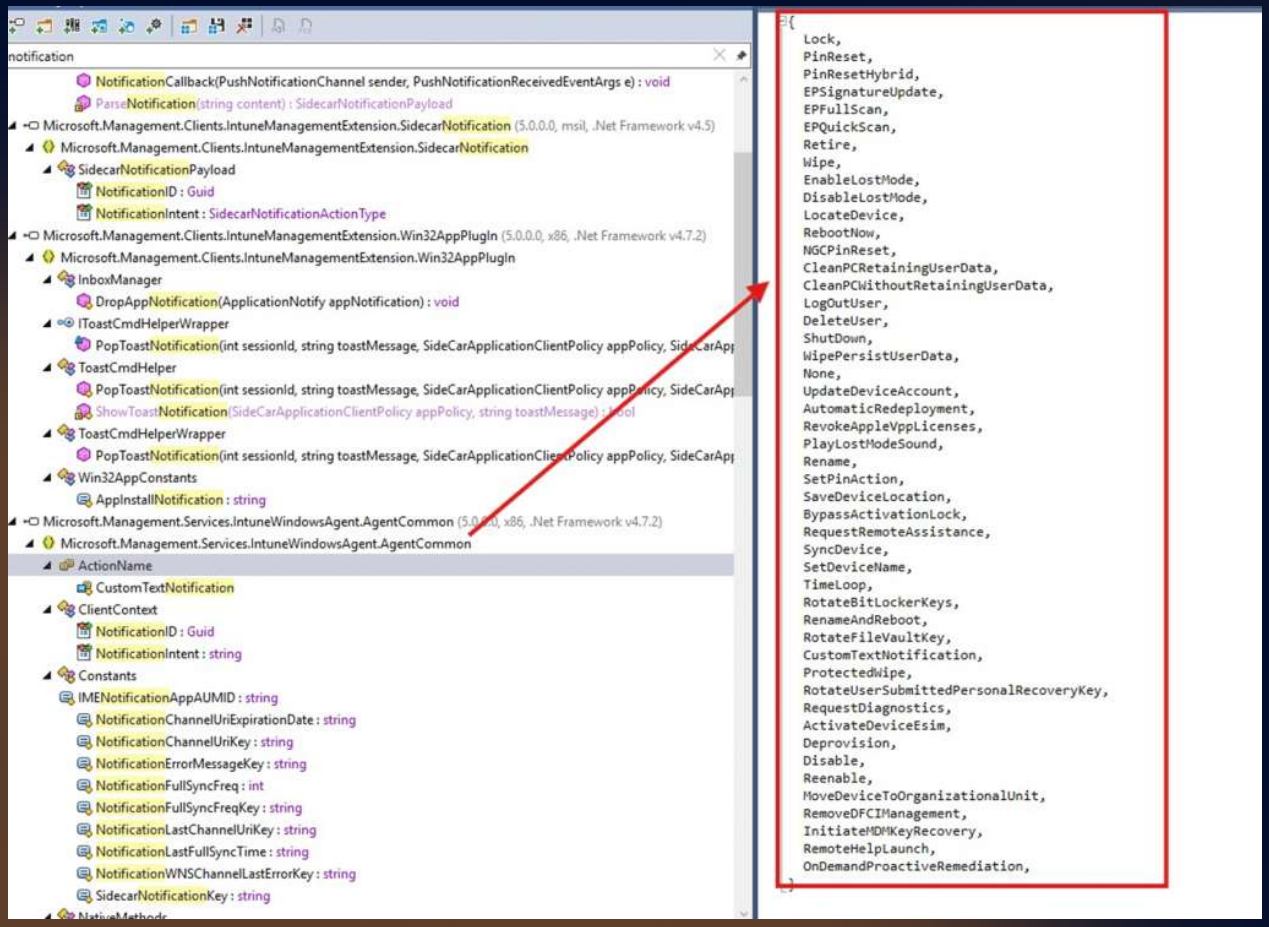
IC3!


Available
Busy
Do not disturb
Be right back
Appear away
Appear offline
Teams and its "connection status" (Online/Offline/Away) is well known
using System;
#nullable disable
namespace Microsoft.Management.Services.IntuneWindowsAgent.IC3;
public class IC3ConnectionStats
{
public int SuccessfulConnections { get; }
public int FailedConnections { get; }
public int ReconnectAttempts { get; }
public int DisconnectAttempts { get; }
public IC3ConnectionStats(
int successfulConnections,
int failedConnections,
int reconnectAttempts,
int disconnectAttempts)
{
this.SuccessfulConnections = successfulConnections;
this.FailedConnections = failedConnections;
this.ReconnectAttempts = reconnectAttempts;
this.DisconnectAttempts = disconnectAttempts;
}
public override string ToString()
{
return string.Join(Environment.NewLine, "SuccessfulConnections=" + this.SuccessfulConnections.ToString(), "FailedConnections=" + this.Failed
}
}
Devices | Windows > Windows | Windows devices >
-57767
Search
Overview
Manage
Properties
Monitor
Device inventory
Device query
Hardware
Discovered apps
Retire
Wipe
Delete
Remote lock
Sync
Essentials
Device name : WVI
Management name : rudyooms_Windows_9/29/2025_5:38 AM
Ownership : Corporate
Serial number : 2193-6
Phone number : ---
Device manufacturer : Microsoft Corporation
Online Status :
Mockup
What if this is going to show up in Intune?



IC3 is a
replacement/improvement/addition
(or how you want to call it) for
Windows Notification Services

IC3!



With it ALL ... ALL remote actions in Intune...
(also the ones living in MDM world)

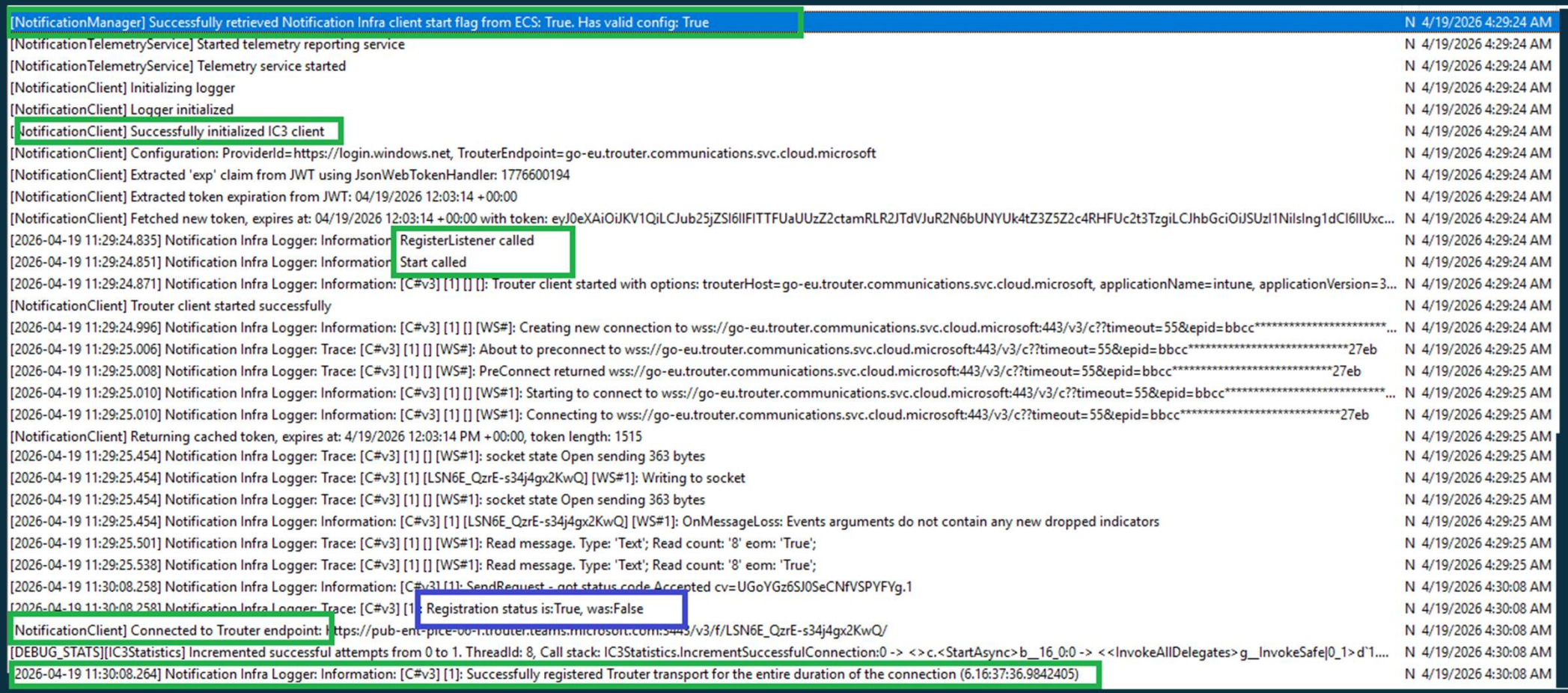
Can move over to the IME and could become instant!

With it the IME becomes PUSH instead of PULL

```
public enum SidecarNotificationActionType
{
    [EnumMember] None,
    [EnumMember] RemoteAssist,
    [EnumMember] Win32AppWorkload,
    [EnumMember] EndpointAnalytics,
    [EnumMember] RunRemediation,
    [EnumMember] IntunePivot,
    [EnumMember] ComplianceScripts,
    [EnumMember] WindowsRemoteHelpUnattended,
}
```

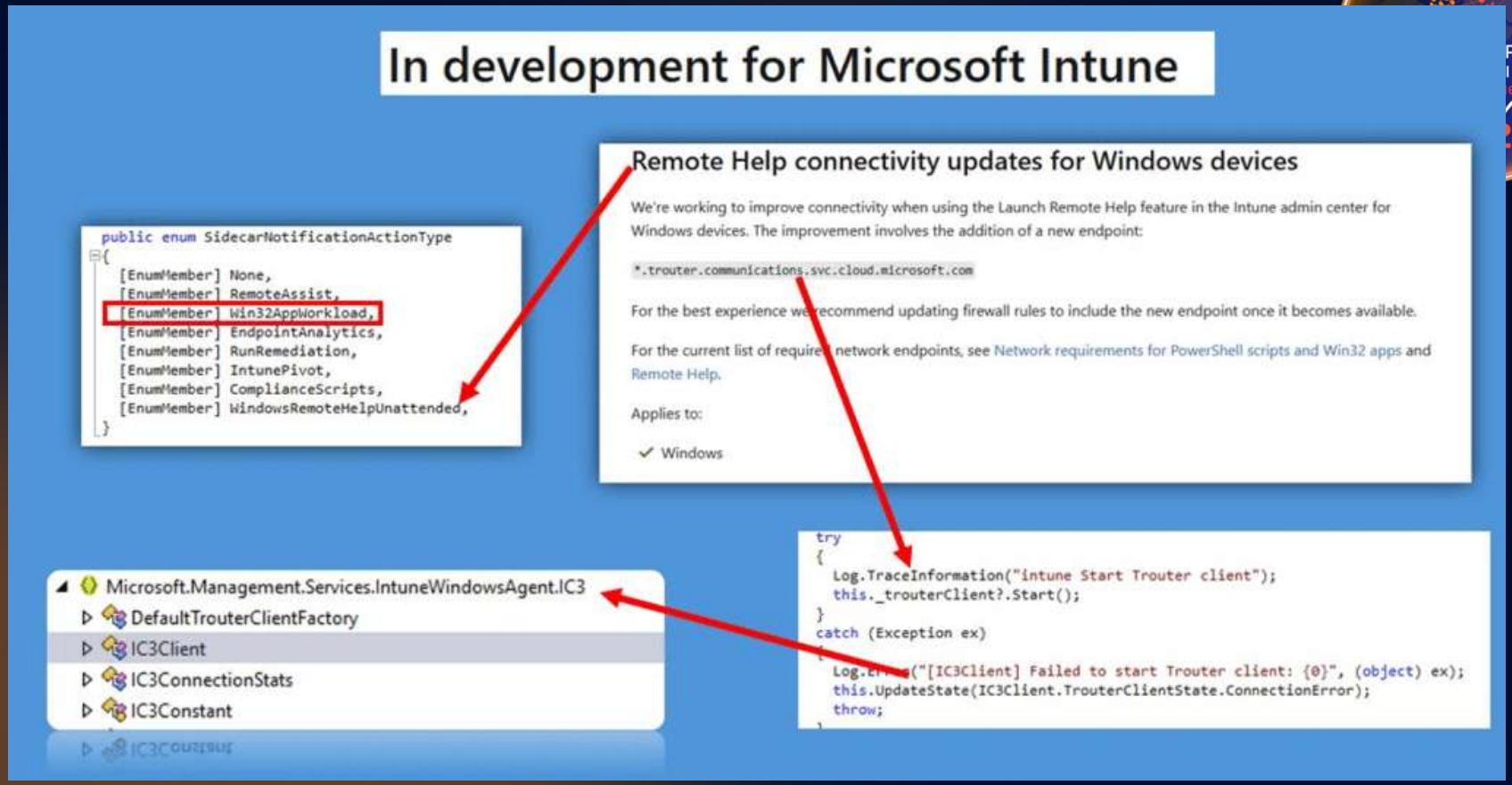
IC3

When IC3 starts working.. The NotificationInfra.log will start showing cool stuff!!



Which is way better then the WNS event log in Windows… Which didn’t show anything useful

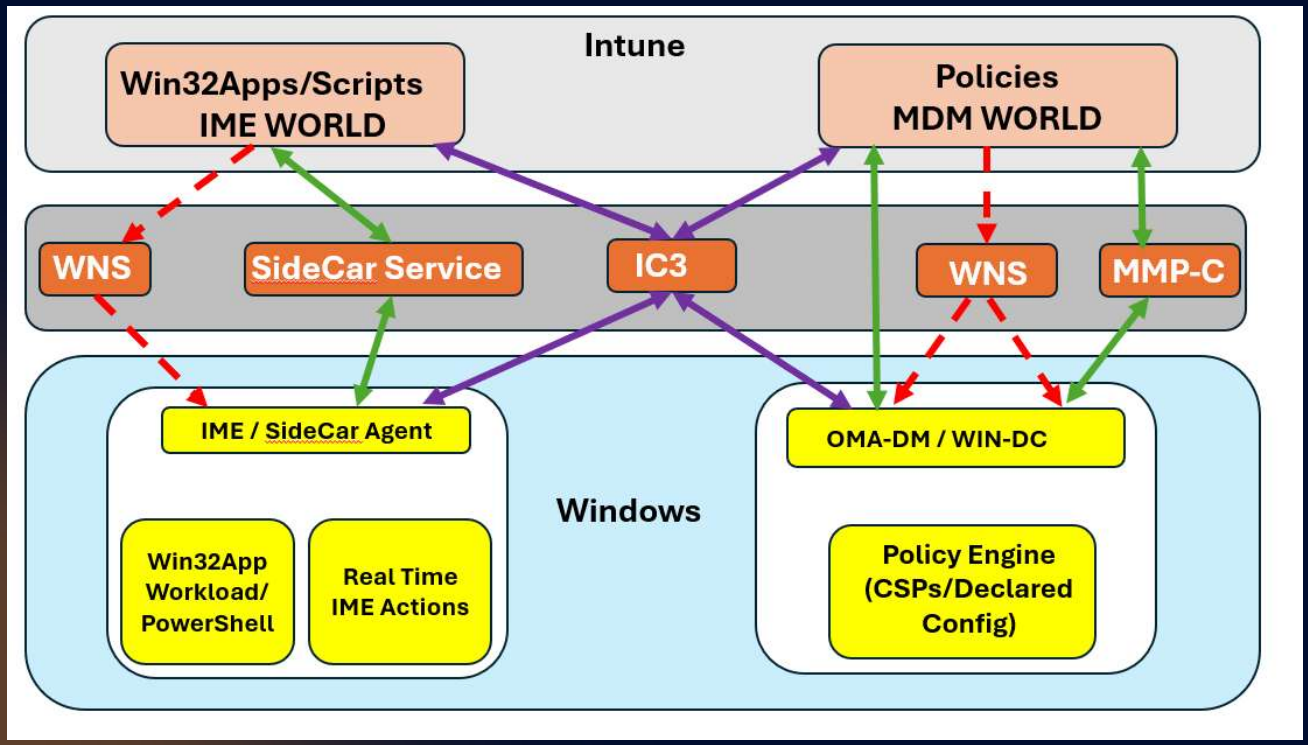
IC3



Remote Help is the first one to start using IC3.
Ensure you allow the new endpoint!!!

Everything Starts Coming Together

Say goodbye to the IME Timers



Summary

- A change in Intune will result in a WNS push to wake up the device to receive the policy
- WNS is a blackbox!
- **Intune today**: Policies apply on check-in, with lots of round trips. (get-set-get)
- Drift goes unnoticed until the next sync.
- **MMP-C fixes this** by enforcing the declared state locally.
- Devices use a **linked dual enrollment**: OMA-DM + MMP-C.
- One configuration document replaces dozens of SyncML transactions. (get-set)
- The IME is also evolving with IC3
- With IC3, Microsoft could ditch their dependency on WNS.

MODERN ENDPOINT MANAGEMENT SUMMIT 2026