**Batch Information:**

* **Batch Start Date:** 2025-08-04
* **Batch Name:** WiproNGA\_DWS\_B5\_25VID2550
* **First Name:** Shrawani
* **Last Name:** Balwadkar
* **User ID:** 34932
* **Batch ID:** B5-25VID2550

**TOPICS**

* Interactive and Non-Interactive Applications
* Required and Available App Assignments
* Groups, Dynamic Queries, Users
* Process Flow for an Application on Windows Client via IME Service
* Registries with Respect to LOB and Win32Apps LOB (Line-of-Business) Apps
* Specific Registries with Application GUID
* Log File Locations
* Company Portal
* How to Sync Once App Assignments are Done
* Breakdown of Events in Log Files

**Microsoft Intune – Application Deployment Documentation**

**1. Interactive and Non-Interactive Applications**

**Interactive Applications**

Interactive applications are those that require some form of user input or interaction during installation. This could be accepting license terms, clicking "Next" buttons, selecting features, or entering user-specific information.

* These applications are typically deployed as **Available** apps so users can initiate installation via the Company Portal.
* Interactive installs can lead to deployment failures if assigned as **Required**, especially if they attempt to prompt the user in the background session.

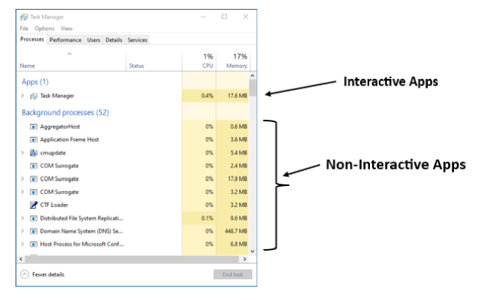
**Example:** Adobe Reader without silent switches, custom setup wizards.

**Non-Interactive Applications**

These are applications that can be installed silently without any user input, using command-line switches such as /quiet, /silent, or /norestart.

* Ideal for enterprise deployment using **Required** assignments.
* Ensure that proper exit codes and detection rules are used to validate successful installation.

**Example:** Google Chrome Enterprise using msiexec /i chrome.msi /quiet /norestart

****

**2. Required and Available App Assignments**

**Required Apps**

* Automatically pushed to target devices or users.
* No manual interaction required.
* Ideal for compliance or business-critical apps.
* Deployed in the background via the **Intune Management Extension (IME)**.

**Available Apps**

* These apps are not auto-installed but made visible in **Company Portal** for self-service installation.
* Best used for optional or productivity tools.
* Assigned to users (not devices) for flexibility and visibility in the portal.

**3. Groups, Dynamic Queries, Users**

**Groups**

* Azure AD Groups (User/Device-based) are used to scope deployments.
* Static Groups: Manually add users/devices.
* Dynamic Groups: Automatically include users/devices based on rules.

**Dynamic Queries**

* Used to auto-populate groups based on device/user attributes.  
  Example query for devices running Windows 10 21H2:
* (device.deviceOSType -eq "Windows") and (device.deviceOSVersion -contains "10.0.19044")

**Users**

* Used in app targeting, especially for **Available** deployments.
* Users must be licensed with Intune and enrolled in MDM.
* Also relevant for Conditional Access and Self-Service scenarios.

**4. Process Flow for an Application on Windows Client via IME Service**

**Step-by-Step Breakdown:**

1. **Polling**
   * The Intune Management Extension (IME) polls the Intune service every **60 minutes** for updates.
   * This polling interval is fixed for Required apps but can be triggered manually via **Sync**.
2. **Application Detection**
   * Before installation, IME checks for the presence of the app using **Detection Rules**.
   * Supported detection methods: File, Registry, MSI, Custom Script.
   * If the app is already installed (per detection), IME skips installation.
3. **Application Download and Installation**
   * If detection fails, the app is downloaded from the Intune content delivery network (CDN).
   * Installation command (silent) is executed.
   * Output and status are written to the logs: IntuneManagementExtension.log and AgentExecutor.log.
4. **Post-Installation Detection**
   * IME re-evaluates the detection logic to ensure successful install.
   * If it passes, app status = Success; if not, status = Failed.
5. **Toast Notification**
   * The user may receive a toast message indicating:
     + Installation Successful
     + Installation Failed
     + App Will Install Soon (based on deadline)

**5. Registries with Respect to LOB and Win32Apps LOB (Line-of-Business) Apps**

**Registry Path:**

HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\EnterpriseDesktopAppManagement\<AppGUID>\

* This registry is mostly used by older LOB deployment methods.

**Win32 Apps (Deployed via Intune Win32 Packaging Tool)**

**Registry Path:**

HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\IntuneManagementExtension\Win32Apps\<AppID>\

* **Common registry values:**
  + InstallStatus: 0 = Not started, 1 = Installing, 2 = Success, 3 = Failed.
  + DetectionState: 0 = Not Detected, 1 = Installed.
  + ResultCode: 0 = Success, any non-zero value indicates a specific error.

**6. Specific Registries with Application GUID**

For each deployed app, a unique **App GUID** or **AppID** is created under:

HKLM\SOFTWARE\Microsoft\IntuneManagementExtension\Win32Apps\<AppGUID>\

Key registry values to check:

* **ResultCode**: Returns exit code from installer. 0 = Success, other values indicate failure.
* **RestartRequired**: 1 indicates a reboot is needed post-install.
* **DetectionState**: 1 if app is detected as installed, 0 if not.
* **LastModifiedTimeUtc**: Timestamp for last app status change.

****

**7. Log File Locations**

Logs are critical for troubleshooting installation issues.

| **Log File** | **Location** | **Purpose** |
| --- | --- | --- |
| IntuneManagementExtension.log | C:\ProgramData\Microsoft\IntuneManagementExtension\Logs\ | Main log for app processing |
| AgentExecutor.log | Same folder | Shows script execution commands |
| DetectionScript.log | Same folder | Output of detection scripts |
| AppIntentEval.log | Same folder | App evaluation status per assignment |

**8. Company Portal**

Company Portal is the client-facing app that allows users to:

* View and install **Available** apps.
* Initiate **device sync**.
* See **status messages** for apps.
* Review **device compliance** and **assigned resources**.

It’s available on the Microsoft Store and can also be deployed via Intune.

**9. How to Sync Once App Assignments are Done**

**Manual Sync from Device Settings**

1. Go to: Settings > Accounts > Access Work or School > [Click your org] > Info
2. Click on **Sync** button.

**Sync via Company Portal**

1. Open **Company Portal** > Go to **Settings** > Click on **Sync**.

**Sync from PowerShell**

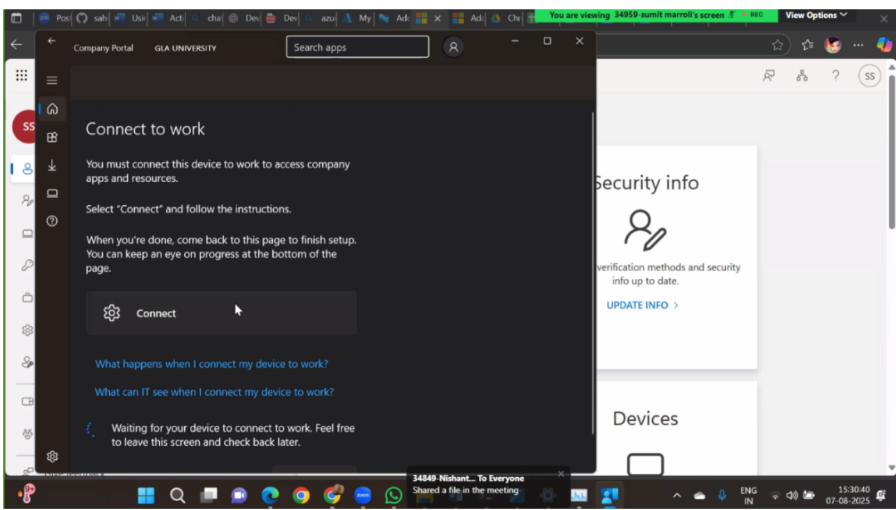
* Command to verify AAD join and MDM status:

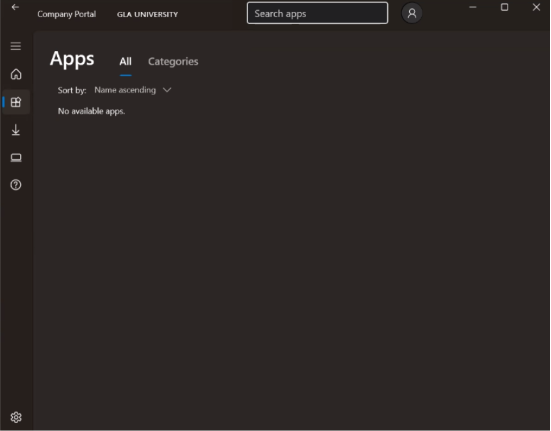
dsregcmd /status

To restart sync (indirect method):

Restart-Service IntuneManagementExtension

This will force the IME to re-evaluate assigned apps/policies.





**10. Breakdown of Events in Log Files**

**IntuneManagementExtension.log**

* Logs detection, install, and result phases.
* Key entries to look for:
  + Detection rule returned false
  + App is applicable for install
  + Install command executed
  + Exit code = 0
  + Detection rule returned true

**AgentExecutor.log**

* Logs script-related operations (install/uninstall/detection scripts).
* Shows full command-line, start time, and exit code.

**DetectionScript.log**

* Logs output of detection scripts (if used).
* Helpful when custom PowerShell logic is used to detect application presence.