



UNIVERSAL APP INSTALLER - QUICK REFERENCE CARD

Print this page for quick reference during deployment



Most Common Commands

```
powershell

# Test installer directly
.\Universal-AppInstaller.ps1 -AppName "AppName" -InstallerFileName "app.exe" `
    -InstallerType "EXE" -InstallArguments "/S" `
    -DetectionMethod "File" -DetectionPath "C:\Program Files\App\app.exe"

# Dry run test
.\Orchestration-Master.ps1 -Phase Phase4 -DryRun

# Run Phase 4 only
.\Orchestration-Master.ps1 -Phase Phase4

# Check exit code
$LASTEXITCODE

# View recent log
Get-Content "C:\ProgramData\OrchestrationLogs\Apps\Install-AppName_*.log" -Tail 50
```



Exit Codes

Code	Meaning
0	✔ Success
1	✘ General failure
2	✘ Installer not found
4	✘ Install failed
5	✘ Validation failed
10	✔ Already installed

Configuration Template

```
powershell

@{
  TaskID = "APP-XXX"
  TaskName = "Install AppName"
  ScriptPath = "Scripts\Universal-AppInstaller.ps1"
  Enabled = $true
  Timeout = 600
  RunAs = "SYSTEM"
  RequiresReboot = $false
  AllowRetry = $true
  Critical = $false
  Description = "Installs AppName"
  Parameters = @{
    AppName = "Application Name"
    InstallerFileName = "installer.exe"
    InstallerType = "AUTO"
    InstallArguments = "/S"
    DetectionMethod = "File"      # Registry, File, AppX, Package
    DetectionPath = "C:\Path\file.exe"
    DetectionValue = ""          # Optional
    RequiredVersion = ""         # Optional
  }
}
```

Detection Methods

Method	DetectionPath Example	Use When
Registry	HKLM:\SOFTWARE\App	App writes registry
File	C:\Program Files\App\app.exe	Predictable install path
AppX	Microsoft.AppName	Modern Store apps
Package	AppName*	Package manager apps

Common Silent Switches

Installer	Switch	Example Apps
MSI	<code>/quiet /norestart</code>	7-Zip, PuTTY
NSIS	<code>/S</code>	Notepad++, WinSCP
Inno Setup	<code>/VERYSILENT /SUPPRESSMSGBOXES /NORESTART /SP-</code>	Git, TreeSize
Adobe	<code>/sAll /rs /msi EULA_ACCEPT=YES</code>	Adobe Reader

File Locations

Scripts\Universal-AppInstaller.ps1 ← The script
Installers\Apps\ ← Your installers
C:\ProgramData\OrchestrationLogs\Apps\ ← App logs

Quick Troubleshooting

Problem	Quick Fix
Installer not found	Check filename & location
Install fails	Test manually: <code>.\installer.exe /S</code>
Detection fails	Verify path: <code>Test-Path "path"</code>
Already installed keeps running	Fix detection method

Testing Checklist

- ☐ Test installer manually
- ☐ Test Universal Installer script
- ☐ Test in dry run mode
- ☐ Test Phase 4 execution
- ☐ Check logs for errors
- ☐ Verify app works

Key Paths

```
powershell
```

```
# Check if installer exists
```

```
Test-Path ".\Installers\Apps\installer.exe"
```

```
# Check if app installed (File)
```

```
Test-Path "C:\Program Files\App\app.exe"
```

```
# Check if app installed (Registry)
```

```
Test-Path "HKLM:\SOFTWARE\App"
```

```
Get-ItemProperty "HKLM:\SOFTWARE\App"
```

```
# View logs
```

```
Get-ChildItem "C:\ProgramData\OrchestrationLogs\Apps\"
```

Detection Examples

Registry Detection:

```
powershell
```

```
DetectionMethod = "Registry"
```

```
DetectionPath = "HKLM:\SOFTWARE\7-Zip"
```

```
DetectionValue = "Path" # Optional
```

File Detection:

```
powershell
```

```
DetectionMethod = "File"
```

```
DetectionPath = "C:\Program Files\Notepad++\notepad++.exe"
```

```
RequiredVersion = "8.6.9" # Optional
```

AppX Detection:

```
powershell
```

```
DetectionMethod = "AppX"
DetectionPath = "Microsoft.WindowsTerminal"
```

Quick App Add

60-Second App Addition:

1. Copy installer to `Installers\Apps\`
2. Test: `.\installer.exe /S`
3. Note install location
4. Copy config template
5. Update: TaskID, AppName, InstallerFileName, InstallArguments, DetectionPath
6. Test: `.\Orchestration-Master.ps1 -Phase Phase4 -DryRun`
7. Deploy: `.\Orchestration-Master.ps1 -Phase Phase4`

Documentation Files

File	Purpose
QuickStart.md	5-min setup
Documentation.md	Complete reference
Sample.ps1	20+ examples
Architecture.md	System design
Checklist.md	Implementation steps
README.md	Overview

When to Use

Use Universal Installer	Use Custom Script
<input checked="" type="checkbox"/> Standard EXE/MSI	<input checked="" type="checkbox"/> Microsoft Office
<input checked="" type="checkbox"/> Simple detection	<input checked="" type="checkbox"/> SQL Server

Use Universal Installer	Use Custom Script
✔ 7-Zip, Notepad++	✗ Chrome policies
✔ VLC, Adobe Reader	✗ Complex config
✔ 80% of apps	✗ 20% of apps

Pro Tips

1. **Always test manually first**
2. **Use MSI when available**
3. **Prefer File over Registry detection**
4. **Set realistic timeouts**
5. **Enable retry for all apps**
6. **Document custom switches**
7. **Keep installer names consistent**
8. **Review logs regularly**

Common Mistakes

✗ Don't:

- Forget to copy installer to Apps folder
- Misspell installer filename
- Use wrong silent switch
- Skip testing manually first
- Ignore log files

✔ Do:

- Test each step incrementally
- Review logs after each run
- Use descriptive TaskNames

- Document special requirements
 - Keep installers organized
-

Success Metrics

Target Goals:

- Success Rate: >95%
 - Install Time: <5 min avg
 - Detection: 100% reliable
 - Support Tickets: <5% of deployments
-

Quick Links

Package Contents:

- Universal-AppInstaller.ps1 (879 lines)
- Phase4-Applications-Sample.ps1 (645 lines)
- Documentation.md (804 lines)
- QuickStart.md (461 lines)
- Architecture.md (550 lines)
- Checklist.md (600+ lines)
- README.md (510 lines)

Total: 130 KB, 4,500+ lines

Copy-Paste Templates

MSI App:

powershell

```
@{
    TaskID = "APP-XXX"; TaskName = "Install AppName"
    ScriptPath = "Scripts\Universal-AppInstaller.ps1"
    Enabled = $true; Timeout = 600; RunAs = "SYSTEM"
    RequiresReboot = $false; AllowRetry = $true; Critical = $false
    Parameters = @{
        AppName = "AppName"; InstallerFileName = "app.msi"
        InstallerType = "MSI"; InstallArguments = "/quiet /norestart"
        DetectionMethod = "Registry"; DetectionPath = "HKLM:\SOFTWARE\App"
    }
}
```

EXE App:

```
powershell

@{
    TaskID = "APP-XXX"; TaskName = "Install AppName"
    ScriptPath = "Scripts\Universal-AppInstaller.ps1"
    Enabled = $true; Timeout = 600; RunAs = "SYSTEM"
    RequiresReboot = $false; AllowRetry = $true; Critical = $false
    Parameters = @{
        AppName = "AppName"; InstallerFileName = "app.exe"
        InstallerType = "EXE"; InstallArguments = "/S"
        DetectionMethod = "File"; DetectionPath = "C:\Program Files\App\app.exe"
    }
}
```

Keep this card handy for quick reference during deployments!

© Universal App Installer v1.0 - Part of Windows 11 Enterprise Orchestration Framework