

PowerShell App Deployment Toolkit (PSADT)

Platform Values of the PowerShell App Deployment Toolkit:

1. Complex Scripting Made Simple:

PSADT abstracts away a lot of complex PowerShell scripting by offering **pre-built functions** and **structured templates**.

2. Consistent User Experience:

The toolkit standardizes how installations look and feel to the end-user (e.g., prompts, progress bars, deferral windows).

3. Improved Deployment Success:

PSADT includes logic to handle dependencies, close running applications, and retry failed installations, increasing the success rate.

4. Open Source and Extensible:

PSADT is an open-source project maintained on GitHub. You can **customize**, **extend**, or contribute to it.

PowerShell App Deployment Toolkit Concepts

1. ADT Session Object

ADT = App Deployment Toolkit

- The **ADT Session Object** is the internal object created by the toolkit when it runs. It is the **engine** that powers the framework during execution.

2. Predefined Functions (Toolkit Functions)

- PSADT includes **dozens of predefined functions** for common deployment tasks. These are built into `AppDeployToolkitMain.ps1`

3. Exit Codes

- Exit codes are crucial in deployment environments (e.g., SCCM, Intune) to indicate success or failure.

4. Deployment Modules

This typically refers to the **deployment lifecycle logic** managed by the main script: Deploy-Application.ps1.

Usage of the PowerShell App Deployment Toolkit:

ADT Session (App Deployment Toolkit Session)

- The **ADT session** starts when you run Deploy-Application.ps1.

Running the Script

You run the Deploy-Application.ps1 script to start the deployment. There are three main **modes**:

Mode	What it does
------	--------------

Install	Installs the application
---------	--------------------------

Uninstall	Uninstalls the application
-----------	----------------------------

Repair	Optional: repairs the installation
--------	------------------------------------

Deploying Applications Using Functions

PSADT provides **predefined functions** to handle deployment tasks.

Scripting Deployment Logic

The Deploy-Application.ps1 script is where you **define** the logic for how the application should be installed, uninstalled, or repaired.

PowerShell App Deployment Toolkit (PSADT) – Deployment Template Guide:

Topic	Purpose
Toolkit & Template	Standard framework and starting point
Understanding Deployment Logic	Core script structure and functions
Using Template Script	Run Deploy-Application.ps1 with parameters

Topic	Purpose
Creating New Deployment	Duplicate and adapt folder/script
Customizing Script	Tailor functions, messages, and behavior
Resources	GitHub, blogs, PowerShell documentation

PowerShell App Deployment Toolkit – Folder Structure:

YourApp_Deployment/

```

└─ AppDeployToolkit/          # Core PSADT engine and config
|   └─ AppDeployToolkitMain.ps1 # Main function library (DO NOT EDIT)
|   └─ AppDeployToolkitConfig.xml # Configuration settings (branding, behavior)
|   └─ AppDeployToolkitHelp.ps1 # Optional helper/reference script
|
└─ Files/                    # Application installer and related files
|   └─ YourInstaller.msi/.exe  # EXE, MSI, or other payloads
|
└─ Images/                   # Custom images (used in UI prompts, logos)
|   └─ CompanyLogo.png        # Optional branding
|
└─ Scripts/                  # Additional PowerShell scripts (optional)
|   └─ CustomScript.ps1       # For extra configuration or post-install tasks
|
└─ Configuration/            # Configuration or packaging files (optional)
|   └─ IntuneWinAppUtil.config # If used with Intune packaging tools
|
└─ Deploy-Application.ps1     # Main script to edit (this drives install/uninstall)
└─ Deploy-Application.exe     # Optional GUI launcher for PSADT script
└─ AppDeployToolkitBanner.png # (Optional) Custom banner image for dialogs
└─ ToolkitLogo.png            # (Optional) Logo shown in PSADT dialogs
└─ Readme.txt                 # Optional: document version, notes, etc.
```

Folder/File	Purpose
AppDeployToolkit/	Houses the PSADT engine — don't edit these unless you're customizing core functions.
Files/	Your actual app installer and dependencies go here.
Images/	Logos or custom images shown in pop-up dialogs (optional).
Scripts/	Optional PowerShell scripts for additional tasks.
Configuration/	Packaging tools like IntuneWinAppUtil or detection scripts (optional).
Deploy-Application.ps1	The main script you edit to control install, uninstall, and repair.
Deploy-Application.exe	Optional wrapper that launches the script (often used in SCCM/Intune).