

Introducing Munki 7

Yes, he's talking about Munki again 🤪



Greg Neagle, Oct 2025

But first...

MacSysAdmin 2011

– European Macintosh System Administrators Meeting 2011 –

Pushing Packages with Munki

Greg Neagle

Walt Disney Animation Studios





MacSysAdmin 2012

– European Macintosh System Administrators Meeting 2012 –





Welcome to
MacSysAdmin 2012
Macintosh System Administrators Meeting 2012

MacSysAdmin 2012
Macintosh System Administrators Meeting 2012









MacSysAdmin 2011

– European Macintosh System Administrators Meeting 2011 –

Pushing Packages with Munki

Greg Neagle

Walt Disney Animation Studios



MacSysAdmin 2011
- European Macintosh System Administrators Meeting 2011 -

Pushing Forward with Munki

Greg Neagle
Walt Disney Animation Studios



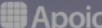
Brought to you by  Apoio

MacSysAdmin 2012
- European Macintosh System Administrators Meeting 2012 -

Advanced Munki

Greg Neagle
Walt Disney Animation Studios



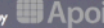
Brought to you by  Apoio

MacSysAdmin 2012
- European Macintosh System Administrators Meeting 2012 -

More Management for Lion and Mountain Lion

Greg Neagle
Walt Disney Animation Studios

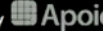


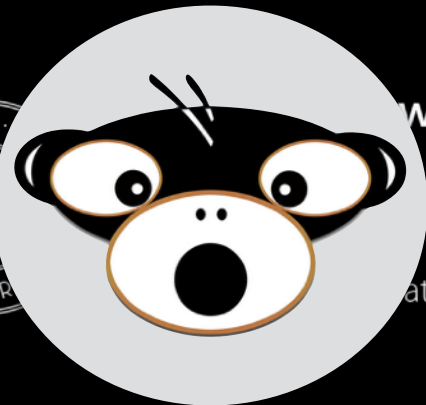
Brought to you by  Apoio



Managing Updates -The Next Level

Greg Neagle

Brought to you by  Apoio



Working with Munki

Greg Neagle
Walt Disney Animation Studios



AutoPkg: Crowd-sourcing Mac packaging and deployment

Greg Neagle
Walt Disney Animation Studios
Tim Sutton
Concordia University



Preference Management with Profiles

Greg Neagle
Walt Disney Animation Studios



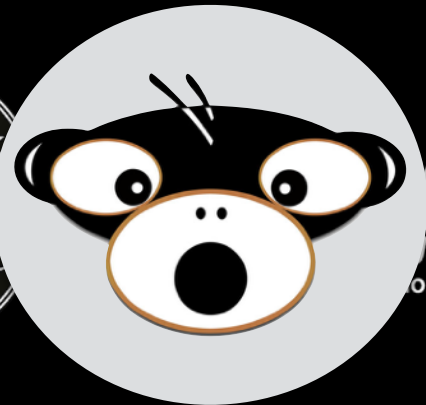
Scripting Opportunities for System Administrators

Greg Neagle
Walt Disney Animation Studios



Command Line Programs All Mac Administrators Should Know

Greg Neagle
Walt Disney Animation Studios
Edward Marczak
Google



Munki 3

Greg Neagle
Walt Disney Animation Studios



Imaging is Dead: What Now?

Greg Neagle
Walt Disney Animation Studios



Autopkg 2020

Greg Neagle



Python and macOS

Greg Neagle, Walt Disney Animation Studios



This one goes to 11

macOS versioning and Big Sur

Greg Neagle, Walt Disney Animation Studios



Munki 5

Changes for 2020

Greg Neagle, Walt Disney Animation Studios

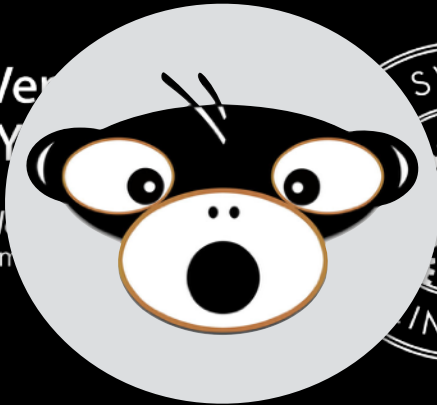
Provisioning in a Pandemic

Greg Neagle
Walt Disney Animation Studios



Munki, Versioning and YAMLL

Greg Neagle
Walt Disney Animation Studios





MUNKI-ING AROUND WITH AI

Greg Neagle



Swiftly Moving On From Python

Greg Neagle

Introducing Munki 7

Yes, he's talking about Munki again 🐼

Greg Neagle, Oct 2025

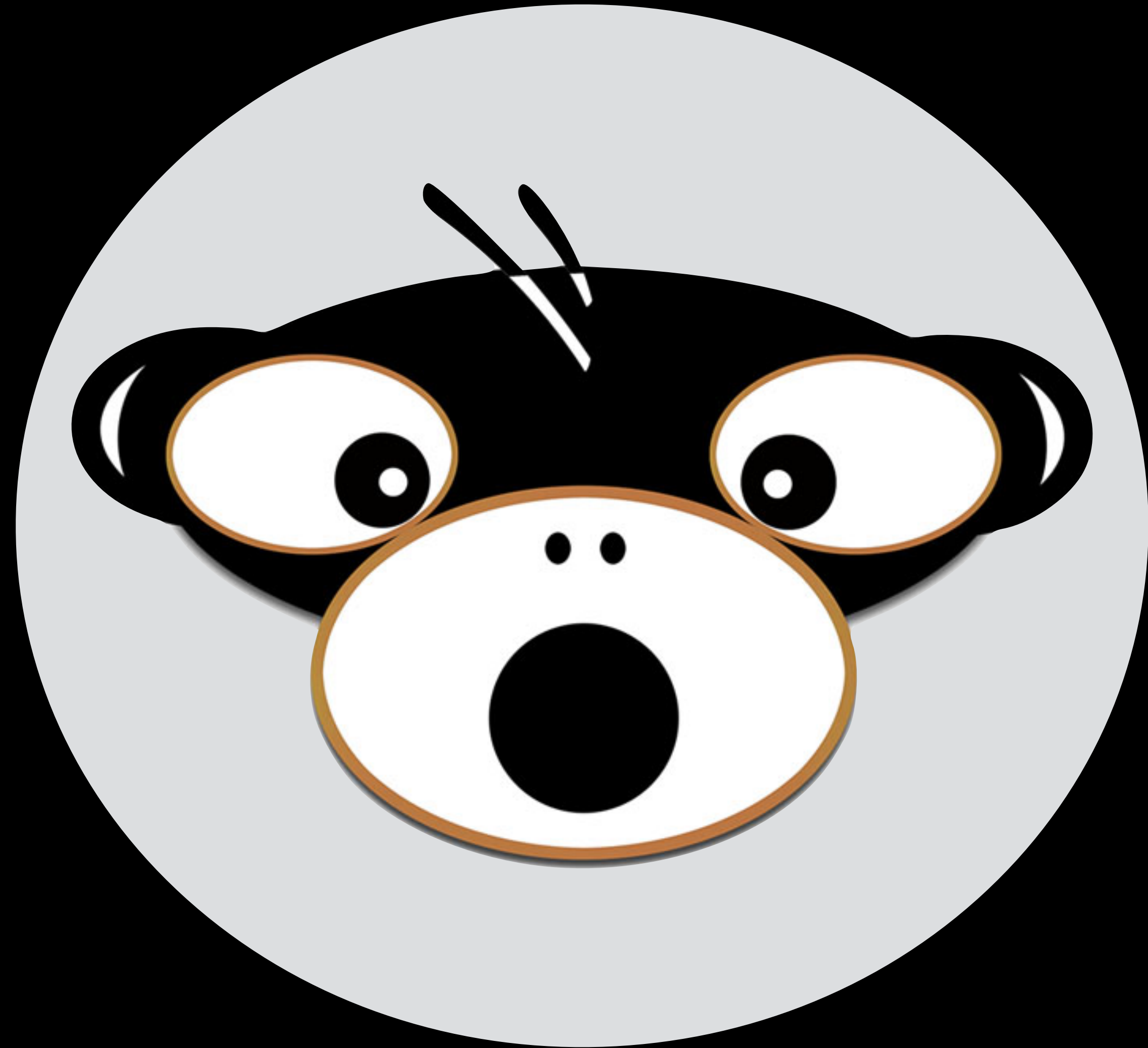


Introducing Munki 7

Yes, he's talking about Munki again 🤪



Greg Neagle, Oct 2025



Search

Software

Categories

My Items

Updates

Managed Software Center

Your source for software for your Mac.



Administration | Cloud Storage | Communication | Developer | Drivers | Editorial | Entertainment | File Sharing | Graphics | Internet | Media
| Production | Productivity | Remote Access | Story | Text Editing | Text Editors | Uncategorized | Utilities | Virtualization | drivers

All items



Adobe Acrobat Reader DC
Productivity - Adobe

INSTALL



Amazon WorkSpaces
Remote Access - Amazon
Installed

REMOVE



Aspera Connect
Utilities - IBM Corp

INSTALL



Atom
Text Editors - GitHub, Inc.

INSTALL



Autodesk Maya 2018
Production - Autodesk

INSTALL



Autodesk Maya 2019
Production - Autodesk

INSTALL



Autodesk Meshmixer
Production - Autodesk

INSTALL



Blender
Production - The Blender Foun...

INSTALL



BlueJeans
Communication - Blue Jeans N...
Installed

REMOVE

https://github.com/munki/munki

github.com

munki / munki

Type / to search

<> Code

Issues 74

Pull requests 4

Discussions

Actions

Projects

Wiki

Security

Insights

Settings

munki

Public

Edit Pins

Unwatch 206

Fork 350

Star 3.3k

main

11 Branches

120 Tags

Go to file

Add file

<> Code

HunterStanton

Fix path traversal in load_page (#1257)

7c22c18 · last month

4,230 Commits

.github	Various typo fixes and spelling corrections (#907)	6 years ago
code	Fix path traversal in load_page (#1257)	last month
launchd	Bump launchd version to match next release	last year
.gitattributes	gitattributes: add .gitattributes for munki-python (#1112)	4 years ago
.gitignore	Updated .gitignore for munkishim	2 years ago
.travis.yml	fix: Add basic PEP8 back into travis ci (#758)	8 years ago
CONTRIBUTING.md	Create CONTRIBUTING.md	5 years ago
LICENSE.md	Replace http://www.apache.org/licenses/LICENSE-2.0 wit...	10 years ago
README.md	Update README.md	2 years ago

README

Contributing

License

Munki

Managed software installation for macOS

About

Managed software installation for macOS —

[www.munki.org/munki/](#)

Readme

View license

Contributing

Activity

Custom properties

3.3k stars

206 watching

350 forks

Report repository

Releases 98

Munki 6.7 Official Release

Latest

3 weeks ago

+ 97 releases

Packages

No packages published

[Publish your first package](#)

Who's using Munki

Adam Anklewicz edited this page on Apr 24, 2024 · [113 revisions](#)

Edit

New page

What does it mean if an organization is included in this list?

Pages 157

It means someone from the organization publicly at least once gave a presentation or made a blog post about using Munki in some capacity. Though probable, it doesn't necessarily mean the organization as a whole uses Munki or that Munki is used for the entire fleet at that organization. It's certainly possible (though highly unlikely) that someone, representating her or his own organization at a conference, gave an in-depth presentation on Munki but uses Munki only personally at home and not in a work context at all.

Guidelines for adding to the list

- Do not "out" organizations (companies/schools) you know are using Munki but who haven't publicly disclosed they are using Munki
- Post a link to the publicly available source along with the date of the source
- Keep the list alphabetical by organization name
- List first the name of the organization, the date of the link (or the full time period using Munki, if disclosed), and then the source

What do the years mean after an organization's name?

That's when the presentation was given or the blog post posted. Sometimes a range of years is evident based on the contents of the presentation/blog, in which case a range is listed. Organizations tend not to implement a solution like Munki

macOS Sequoia info

macOS Sonoma info

macOS Ventura info

Introduction

- [Getting Started](#)
- [Overview](#)
- [Discussion Group](#)
- [Demonstration Setup](#)
- [Glossary](#)
- [Frequently Asked Questions](#)
- [Contributing to Munki](#)
- [Release Notes](#)

Future Plans

- [Deprecation Notes](#)

Munki 7 (in beta)

- [Introduction](#)

Walt Disney Animation Studios

2017 [Introducing Munki 3](#)

Washburn University

2017 [Standard Software – Current as of 3/13/2017](#)

Wheaton College

2017 [Managed Software Center Available for College-Owned Macs](#)

Willem de Kooning academy / Hogeschool Rotterdam, Netherlands

2020 [munki-and-jamf-school](#)

William Paterson University

2017 [Mac OS Managed Software Center](#)

Yale School of Medicine, Department of Pathology

2017 [Pathology ScopeCams](#)

+ Add a custom footer





Commercial implementations

Home > Omnissa products and platform services > Workspace ONE® UEM | Unified endpoint management



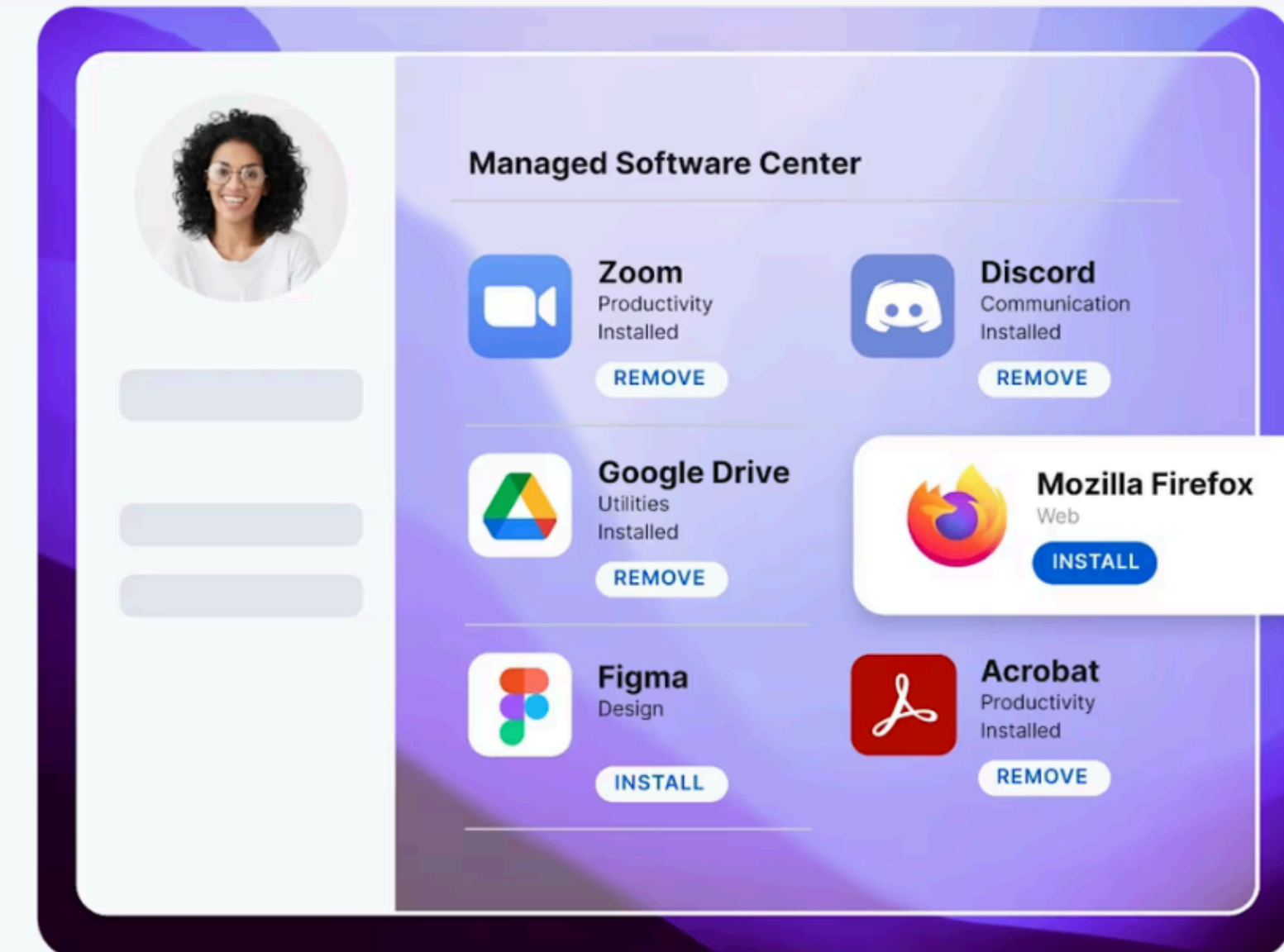
Workspace ONE UEM

Embrace a single, cloud-native product for unified endpoint management—for any device and any use case.



Robust Munki integration

With our [self-hosted Munki integration](#), you can enable your macOS users to self-install software not available in the App Store or native MDM channel, freeing up your valuable time.



Shared apps

Easily add and distribute software (without additional work on your part) with our [shared app directory](#) of up-to-date macOS software.

Custom apps

Upload in-house applications or custom builds. We'll determine whether they can be distributed via MDM, Munki, or both.

Self-serve storefront

Create a self-serve storefront for macOS devices with preselected applications for users to install and uninstall for themselves.

NoPkg support

Run scripts automatically & continuously check for changes in the installed state to determine if they need to run again.

Version pinning

When distributing software, default to update with every new version or use version pinning to select a specific version.

Custom PkgInfo

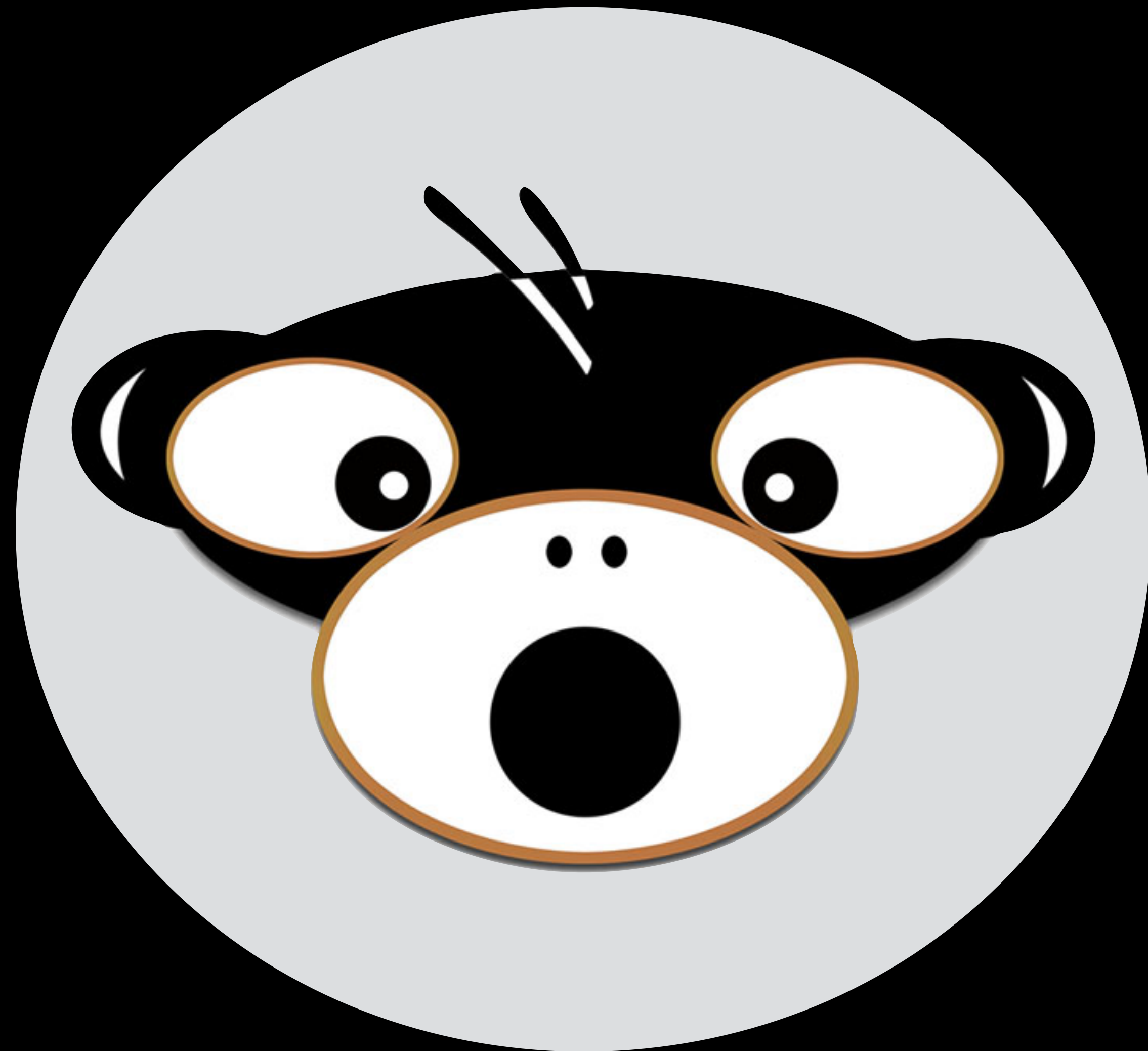
Easily customize the PkgInfo files for your Munki apps. Upload your own PkgInfo file, or edit the existing file to fit your specific needs.



Jamf Auto Update

Automated application packaging and deployment service for Jamf Pro

Jamf Auto Update (previously named dataJAR Auto-Update for Jamf) is a cloud-based service for MSPs that integrates seamlessly with Jamf Pro to provide fully automated application packaging, deployment and updating capabilities for the management of macOS devices.



Munki 6.0 Official Release



gregneagle released this Oct 18, 2022

· [211 commits](#) to Munki6dev since this release



v6.0.0



e739c6e

This is the official release of Munki 6.0, a release adding new features to the Munki tools.

This is the exact same build as Munki 6.0 Release Candidate 1. There have been no changes since that release.

New features

- Munki 6 adds a feature to stage a macOS installer application and have Managed Software Center launch it, temporarily giving the user admin rights if needed. This enables Munki to assist with an upgrade of macOS, and this feature can be used on Intel Macs as well.
- `munkiimport` and `makepkginfo` now support a new `--installer-type` option, which can be used with macOS installers to indicate whether you want to build a `startosinstall` item, a `stage_os_installer`

Munki 6.7 Official Release Latest

 gregneagle released this last month · [0 commits](#) to main since this release  v6.7.0  40364d2

This is the official release of Munki 6.7.0: a bug-fix and enhancement release of the Munki tools.

Fixes and enhancements

- If a process has made a display sleep assertion, skip notification of pending updates (unless a forced log meant to avoid a Munki notification if a user is "presenting". [f88d10e](#) [0a2cb3f](#))
- During bootstrapping, if on A/C power or battery level is greater than 50% on Intel or greater than 30% on ARM, the status app will prevent the display from going to sleep so that bootstrap progress is displayed. Thanks to [@mattmcc](#) ([#1215](#))
- Some fixes for the placement of the background-blurring windows at boot time: [abb511e](#) [7bda74b](#)

But what about Munki 7?!!!



Progress update



Progress update



Progress update



Progress update



About one minute remaining...



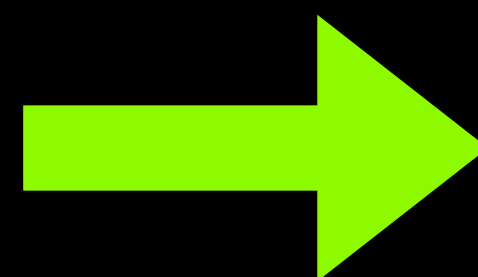
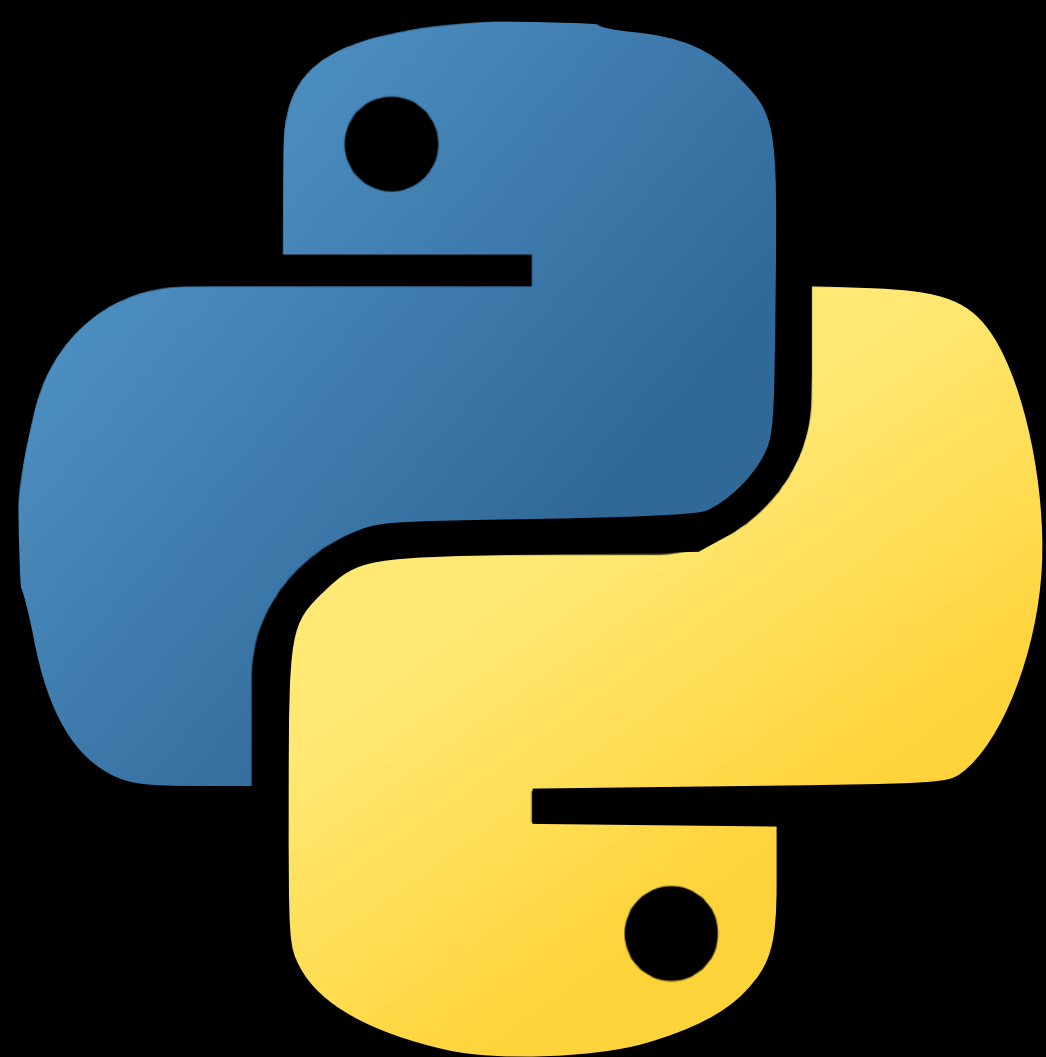
What's changed in Munki 7?

A wide waterfall cascades down a massive, rugged rock face. The rock is light-colored with visible vertical fissures and horizontal strata. The water is white and frothy as it falls. The sky is overcast with soft, grey clouds. In the foreground, the tops of green trees are visible at the bottom and right edges of the frame.

What's new

Command-line tools







bash-3.2\$

bash-3.2\$

Side-effects



Login Items & Extensions

Search



Greg Neagle
Apple Account



Family



Wi-Fi



Bluetooth



Network



Energy



General



Accessibility



Appearance



Menu Bar



Siri



Desktop & Dock



Displays



Spotlight



Wallpaper



Notifications



Sound

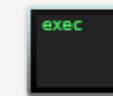
App Background Activity

Apps run in the background after you close them to perform tasks such as checking for updates. Personal data and device sensors can be accessed by apps while they run in the background, if you grant permission.



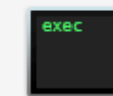
app_usage_monitor

Item from unidentified developer.



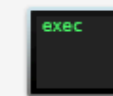
appusaged

Item from unidentified developer.



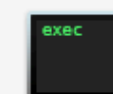
authrestartd

Item from unidentified developer.



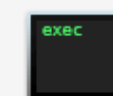
launchapp

Item from unidentified developer.



launchapp

Item from unidentified developer.



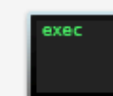
logouthelper

Item from unidentified developer.



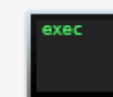
Managed Software Center

1 item



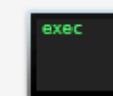
supervisor

Item from unidentified developer.



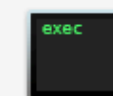
supervisor

Item from unidentified developer.



supervisor

Item from unidentified developer.



supervisor

Item from unidentified developer.





Search



Greg Neagle

Apple Account



Family



Wi-Fi



Bluetooth



Network



Energy



General



Login Items & Extensions

App Background Activity

Apps run in the background after you close them to perform tasks such as checking for updates. Personal data and device sensors can be accessed by apps while they run in the background, if you grant permission.



Managed Software Center

11 items: 6 items affect all users



VirtualBuddyGuest

1 item



Extensions

Extensions add extra functionality to your Mac and apps, and some may run in the background.

By App

By Category



**Books
Sharing**



Freeform



A wide-angle photograph of a coastal scene. In the foreground, white-capped waves are breaking onto a dark, pebbly beach. The middle ground features a long, dark wooden pier extending from the left side towards the horizon. The sea is a deep, dark blue-grey. The sky is filled with large, dark, heavy clouds, with some lighter patches where sunlight is breaking through. The overall mood is dramatic and moody.

One new feature

version_script


```
<key>installs</key>
  <array>
    <dict>
      <key>CFBundleIdentifier</key>
      <string>com.google.Chrome</string>
      <key>CFBundleName</key>
      <string>Chrome</string>
      <key>CFBundleShortVersionString</key>
      <string>140.0.7339.186</string>
      <key>CFBundleVersion</key>
      <string>7339.186</string>
      <key>minosversion</key>
      <string>12.0</string>
      <key>path</key>
      <string>/Applications/Google Chrome.app</string>
      <key>type</key>
      <string>application</string>
      <key>version_comparison_key</key>
      <string>CFBundleShortVersionString</string>
    </dict>
  </array>
```



```
<key>name</key>  
<string>hp-easy-scan</string>  
<key>receipts</key>  
<array>  
  <dict>  
    <key>installed_size</key>  
    <integer>268041</integer>  
    <key>packageid</key>  
    <string>com.hp.EasyScanMac.pkg</string>  
    <key>version</key>  
    <string>1.11.0.210909</string>  
  </dict>  
</array>
```



```
<key>installcheck_script</key>  
<string>#!/bin/sh
```

```
# Grab current version of installed python module  
version="$(python -c 'import  
argparse;print(argparse.__version__)' 2>/dev/null)"  
  
# Compare with the version we want to install  
if [ ${version:-0} \< 1.2.1 ]; then  
    exit 0  
else  
    exit 1  
fi  
</string>
```



```
#!/bin/sh
```

```
read -p "Enter a version to compare: " VERSION
```

```
if [ "${VERSION}" \< "1.2.1" ] ; then
```

```
    echo "YES, ${VERSION} is less than 1.2.1"
```

```
else
```

```
    echo "NO, ${VERSION} is not less than 1.2.1"
```

```
fi
```

```
% /tmp/test.sh
```

```
Enter a version to compare: 1.1
```



```
YES, 1.1 is less than 1.2.1
```



```
#!/bin/sh
```

```
read -p "Enter a version to compare: " VERSION
```

```
if [ "${VERSION}" \< "1.2.1" ] ; then
```

```
    echo "YES, ${VERSION} is less than 1.2.1"
```

```
else
```

```
    echo "NO, ${VERSION} is not less than 1.2.1"
```

```
fi
```

```
% /tmp/test.sh
```

```
Enter a version to compare: 1.3
```



```
NO, 1.3 is not less than 1.2.1
```



```
#!/bin/sh
```

```
read -p "Enter a version to compare: " VERSION
```

```
if [ "${VERSION}" \< "1.2.1" ] ; then
```

```
    echo "YES, ${VERSION} is less than 1.2.1"
```

```
else
```

```
    echo "NO, ${VERSION} is not less than 1.2.1"
```

```
fi
```

```
% /tmp/test.sh
```

```
Enter a version to compare: 1.10
```



```
YES, 1.10 is less than 1.2.1
```



```
#!/bin/sh
echo $(/usr/local/bin/python3 -c \
    'import argparse ; print(argparse.__version__)' \
    2>/dev/null)
```

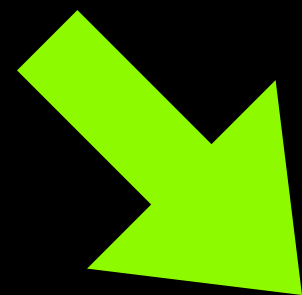
1.1


```
<plist version="1.0">
  <dict>
    <snip>
    <key>version</key>
    <string>1.2.1</string>
    <key>version_script</key>
    <string>#!/bin/sh
echo $(/usr/local/bin/python3 -c \
'import argparse ; print(argparse.__version__)' \
2>/dev/null)
    </string>
  </dict>
</plist>
```




GUI tools





Search

 Software

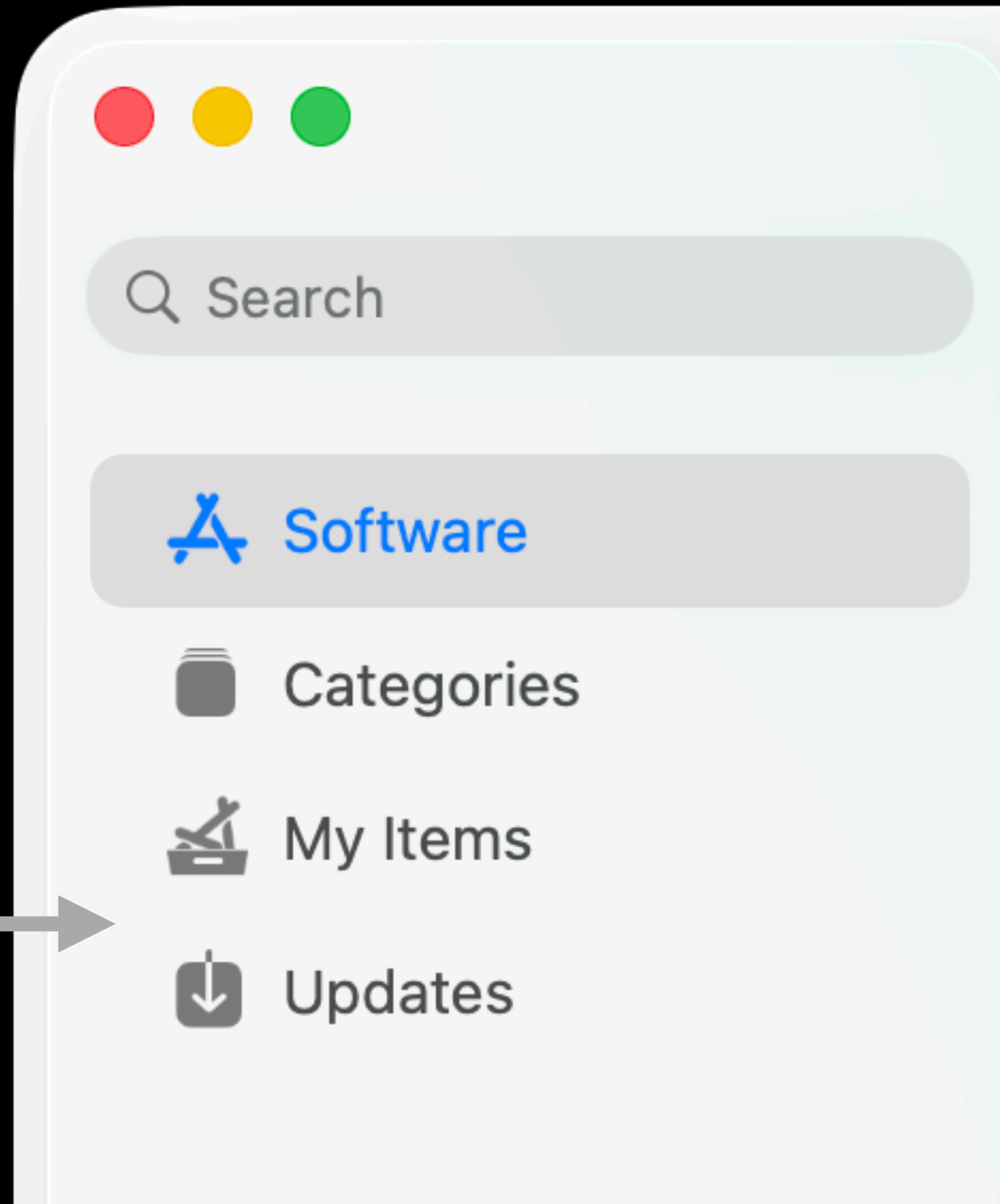
 Categories

 My Items

 Updates

Help | S

I guess that
looks like a
sheet of glass?





Search

 Software

 Categories

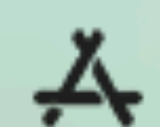
 My Items

 Updates

< Back ⌘ [

↻ Reload Page ⌘ R

🔍 Search ⌘ F

 Software ⌘ 1

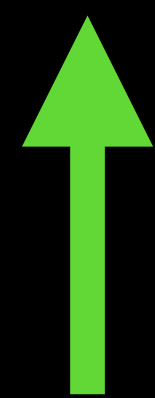
 Categories ⌘ 2

 My Items ⌘ 3

 Updates ⌘ 4











Customizations



Search

Software

Categories

My Items

Updates

1



Help | Support | Change your passwords | All Categories | What's new in macOS Sequoia | Apple | Disney Animation Web

All items



1Password
Utilities - AgileBits Inc.
Installed

REMOVE



1Password for Safari
Utilities - AgileBits Inc.

INSTALL



Adobe Reader
Productivity - Adobe

INSTALL



Amazon WorkSpaces
Remote Access - Amazon

INSTALL



Search

Software

Categories

My Items

Updates

1



Help | Support | Change your passwords | All Categories | What's new in macOS Sequoia | Apple | Disney Animation Web

All items



1Password
Utilities - AgileBits Inc.
Installed

REMOVE



1Password for Safari
Utilities - AgileBits Inc.

INSTALL



Adobe Reader
Productivity - Adobe

INSTALL



Amazon WorkSpaces
Remote Access - Amazon

INSTALL



Search

Software

Categories

My Items

Updates

1



REMOVE



WacomTablet
Drivers - Wacom

INSTALL



Windows App
Remote Access - Microsoft
Installed

REMOVE



Xcode 16
Developer - Apple
Installed

REMOVE



XQuartz
Utilities
Installed

INSTALLED



INSTALL



WDAS Teradici Launcher - XKeys
Remote Access - Walt Disney A...

INSTALL



X-keys Setup!
Utilities - P.I. Engineering, Inc.

INSTALL



Xcode 26 beta 7
Development - Apple
Installed

REMOVE



Zoom
Collaboration - Zoom Video Co...
Installed

REMOVE



Search



Software



Categories

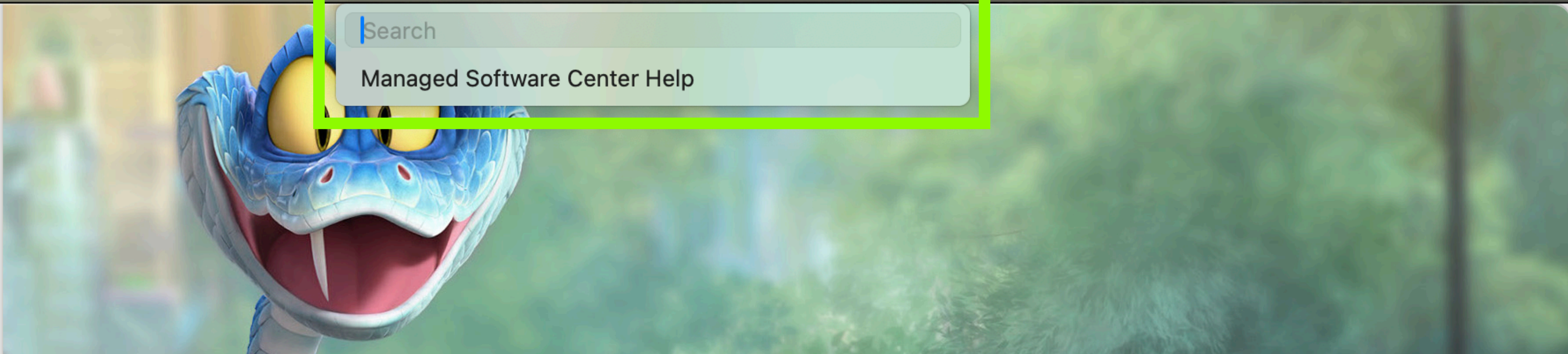


My Items



Updates

1



Search

Managed Software Center Help

[Help](#) | [Support](#) | [Change your passwords](#) | [All Categories](#) | [What's new in macOS Sequoia](#) | [Apple](#) | [Disney Animation Web](#)

All items



1Password
Utilities - AgileBits Inc.
Installed

REMOVE



1Password for Safari
Utilities - AgileBits Inc.

INSTALL



Adobe Reader
Productivity - Adobe

INSTALL



Amazon WorkSpaces
Remote Access - Amazon

INSTALL



Search

Software

Categories

My Items

Updates

1



Help | Support | Change your passwords | All Categories | What's new in macOS Sequoia | Apple | Disney Animation Web

All items



1Password
Utilities - AgileBits Inc.
Installed

REMOVE



1Password for Safari
Utilities - AgileBits Inc.

INSTALL



Adobe Reader
Productivity - Adobe

INSTALL



Amazon WorkSpaces
Remote Access - Amazon

INSTALL

Customizable sidebar



Steve K ng
Z rich, Switzerland



Search

 Software

 Categories

 My Items

 Updates

[Help](#) | [Support](#) | [Change your passwords](#) |

All Categories



[What's new in](#)

[Disney Animation Web](#)

All items



1Password
Utilities - AgileBits Inc.

INSTALL



1Password f
Utilities - Agile

INSTALL



Search

 Software

 Categories

 My Items

 Updates

 Helpdesk



[Help](#) | [Support](#) | [Change your passwords](#) |

All Categories



| [What's new in](#)

| [Disney Animation Web](#)

All items



1Password
Utilities - AgileBits Inc.

INSTALL



1Password f
Utilities - Agile

INSTALL



Search

★ Software

🖥️ My Device

🔍 Helpdesk

📄 My Homepage

📶 Updates

🔧 Clean my Mac



[Help](#) | [Support](#) | [Change your passwords](#) |

All Categories



| [What's new in](#)

| [Disney Animation Web](#)

All items



1Password

Utilities - AgileBits Inc.

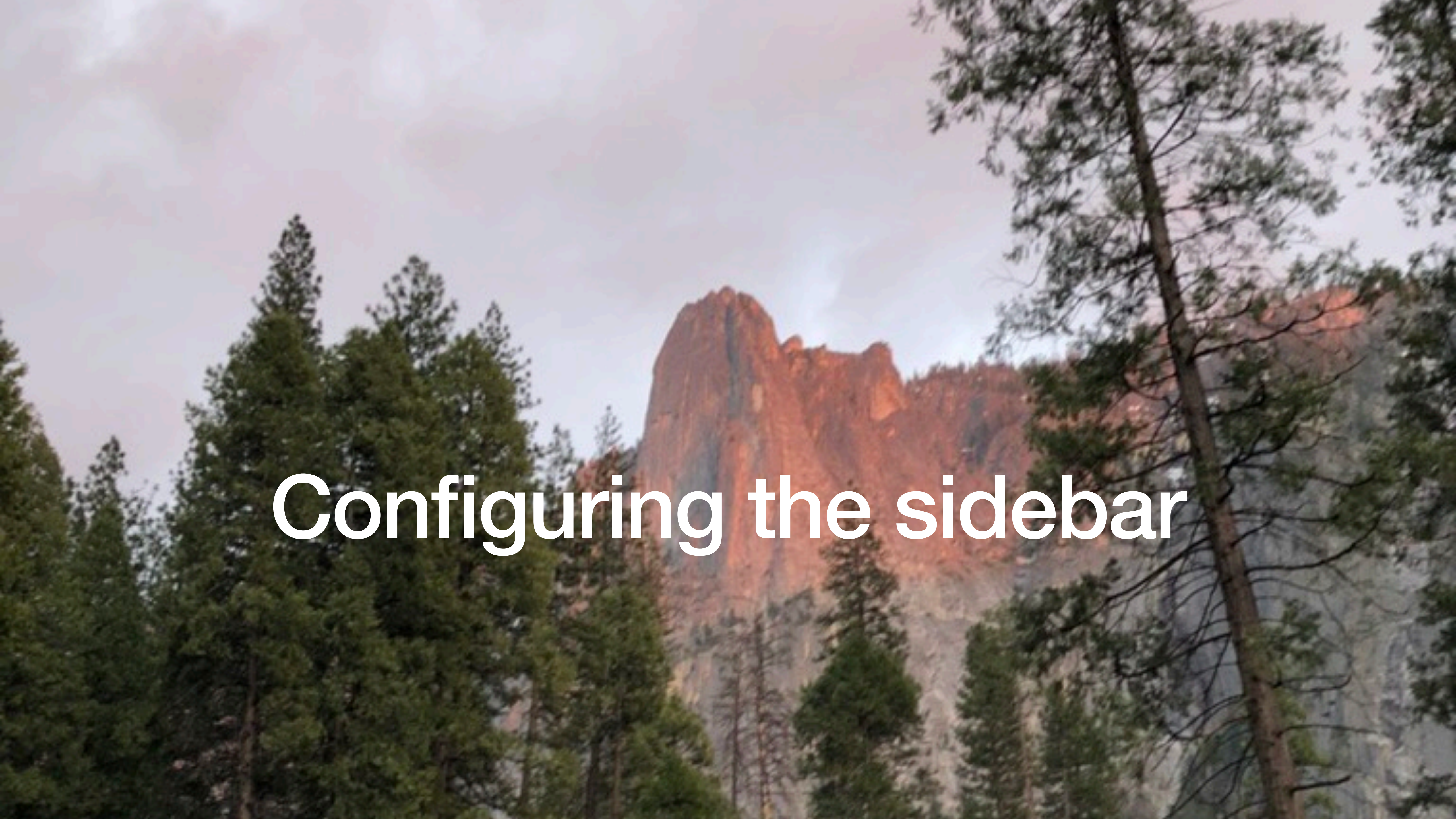
INSTALL



1Password f

Utilities - Agile

INSTALL



Configuring the sidebar

Configuring the sidebar

CustomSidebarItems

- Munki's preferences
(includes preferences delivered via configuration profile)
- A client customization archive file in the Munki repo
(<https://github.com/munki/munki/wiki/Client-Customization>)

github.com

munki / munki

Type / to search

<> Code

Issues 74

Pull requests 4

Discussions

Actions

Projects

Wiki

Security

Insights

Settings

Customizing Managed Software Center sidebar

EditNew page

Greg Neagle edited this page 3 weeks ago · 13 revisions

(New in Munki 7)

NOTE:

In an earlier design of Managed Software Center, icons for Software, Categories, My Items, and Updates were along the top of the window in a toolbar, and there was a sidebar in the main web view.



Admins could customize this "sidebar" by providing a custom sidebar_template.html in client customization resources (see https://github.com/munki/munki/wiki/Client-Customization#sidebar_templatehtml-example).

Pages 157

macOS Sequoia info

macOS Sonoma info

macOS Ventura info

Introduction

- Getting Started
- Overview
- Discussion Group
- Demonstration Setup
- Glossary
- Frequently Asked Questions
- Contributing to Munki
- Release Notes

Future Plans

Demo time!



🔍 Search

☆ Software

🖥️ My Device

🔍 Helpdesk

📄 My Homepage

📶 Updates

🔧 Clean my Mac



Help | Support | Change your passwords | All Categories | What's new in macOS Sequoia | Apple | Disney Animation Web

All items



1Password
Utilities - AgileBits Inc.

INSTALL



1Password for Safari
Utilities - AgileBits Inc.

INSTALL



Adobe Creative Cloud Desktop
Production - Adobe

INSTALL



Adobe Reader
Productivity - Adobe

INSTALL



Amazon WorkSpaces
Remote Access - Amazon

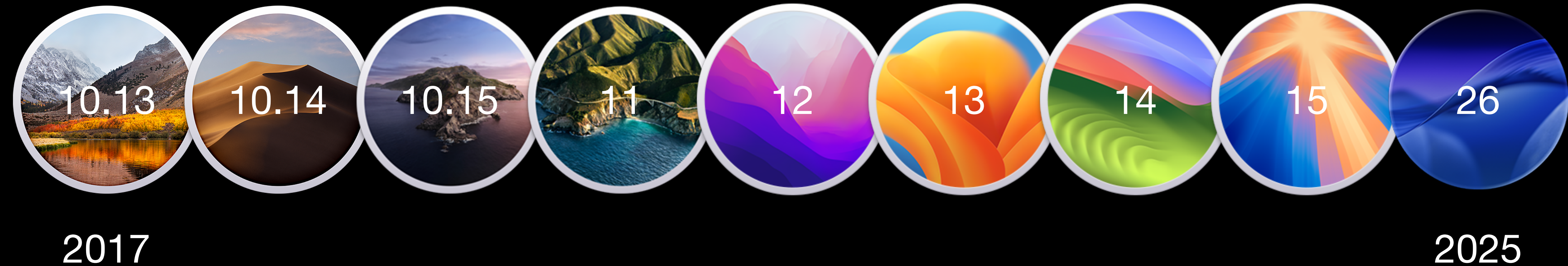


Anyware PCoIP Client
Remote Access - Teradici
Installed

What's going away?



Munki 6.7



Munki 7.0



2025

Munki 7.0



10.15

2019



11



12



13



14



15



26

2025



✗ /usr/local/munki/munki-python

✗ /usr/local/munki/Python.framework

munki-facts

```
#!/usr/local/munki/munki-python
'''Processes python modules in the facts directory and adds the info they
return to our ConditionalItems.plist'''

from __future__ import absolute_import, print_function

import importlib.util
import os
import plistlib
import sys
from xml.parsers.expat import ExpatError

# pylint: disable=no-name-in-module
from CoreFoundation import CFPreferencesCopyAppValue
# pylint: enable=no-name-in-module

def main():
    # pylint: disable=too-many-locals
    '''Run all our fact plugins and collect their data'''
    module_dir = os.path.join(os.path.dirname(__file__), 'facts')
    facts = {}
```




✗ `<key>installer_type</key>`
`<string>startosinstall</string>`

✓ `<key>installer_type</key>`
`<string>stage_os_installer</string>`



macOS Tahoe

Discover

Arcade

Create

Work

Play

Develop

Categories

Updates



Greg Neagle

Results for "macOS Tahoe"

Mac Apps

iPhone & iPad Apps

Filters

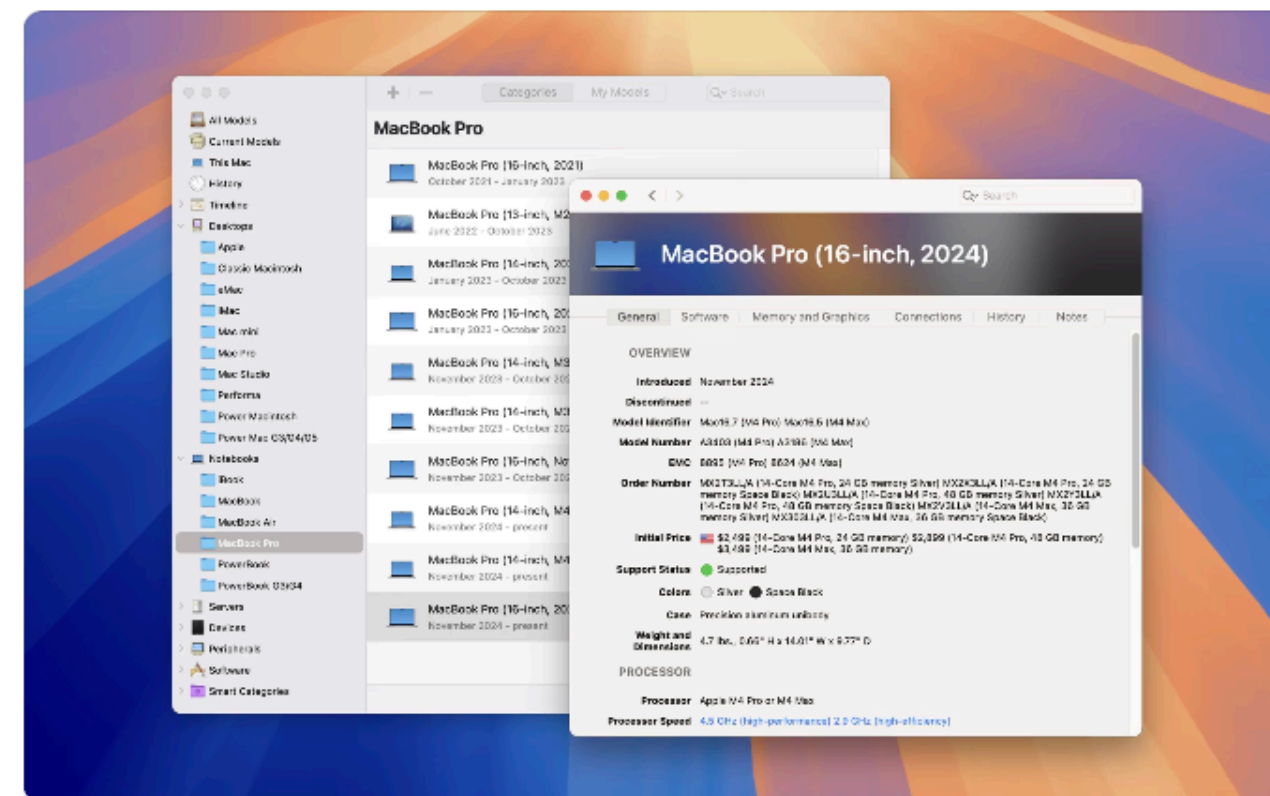


MacTracker

Reference

★★★★★ 606

Get



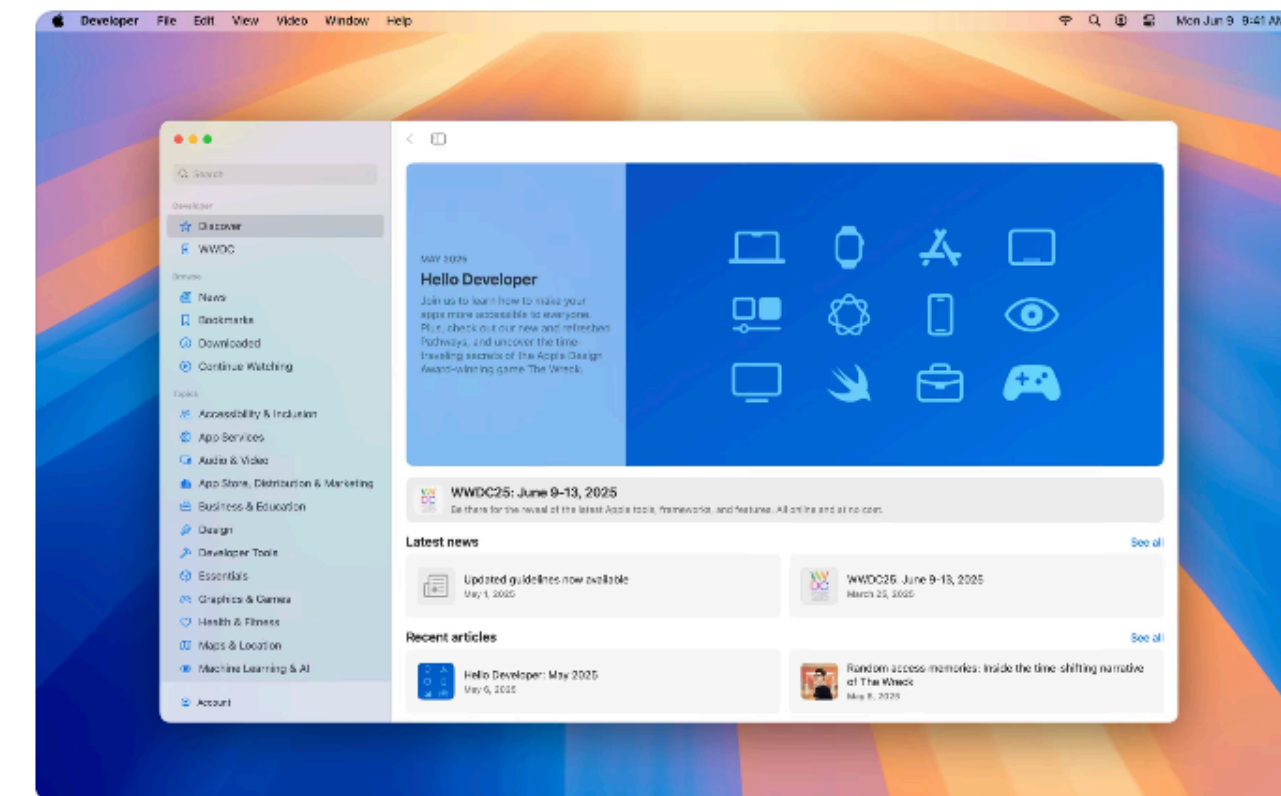
Apple Developer

Developer Tools

★★★★☆ 444

Get

In-App Purchases



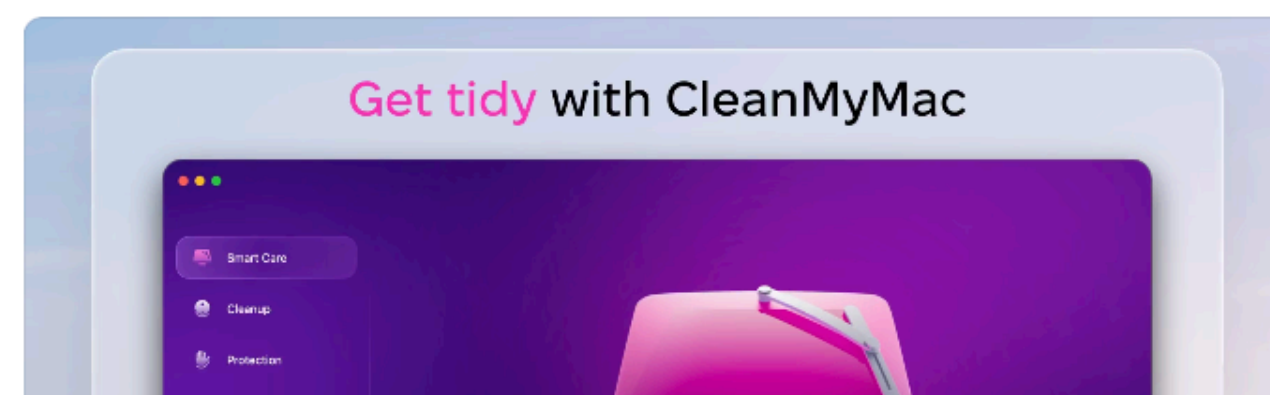
CleanMyMac

Clean & protect your computer

★★★★★ 14K

Get

In-App Purchases

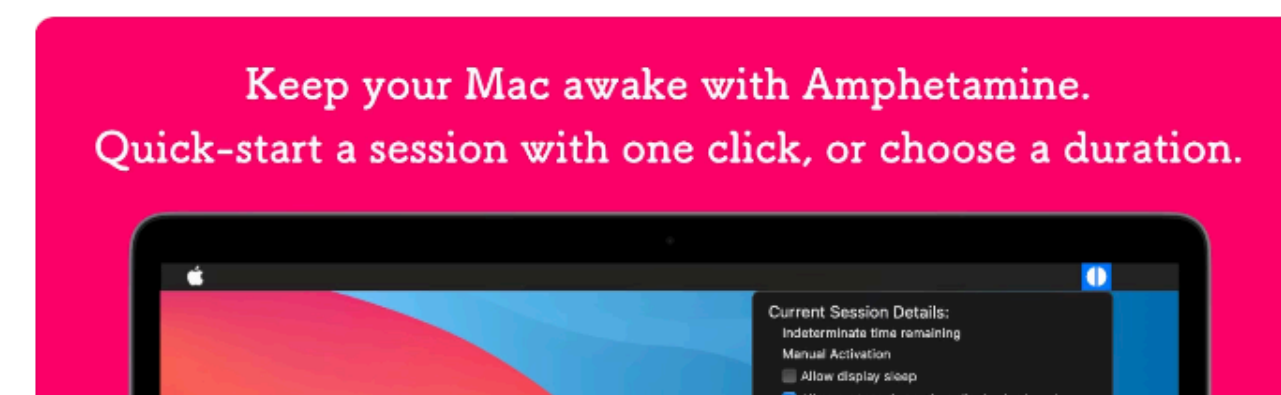


Amphetamine

Powerful keep-awake utility

★★★★★ 2.8K

Get



✗ Installing Apple software updates

Updates

2 pending updates **Restart Required**

UPDATE ALL



macOS Sequoia 15.7

Apple

Version 15.7 • 0.0 KB

Restart Required ⓘ



Command Line Tools for Xcode 26.0

Apple

Version 26.0 • 4.6 GB

The Command Line Tools enable UNIX-style development via Terminal by installing command line...

[More](#)



installer_type:

- ✗ AdobeSetup
- ✗ AdobeUberInstaller
- ✗ AdobeAcrobatUpdater
- ✗ AdobeCS5AAMEEPackage
- ✗ AdobeCS5PatchInstaller
- ✗ AdobeCCPInstaller

uninstall_method:

- ✗ AdobeSetup
- ✗ AdobeUberUninstaller
- ✗ AdobeCS5AAMEEPackage
- ✗ AdobeCCPUUninstaller

✗ Configuration Profiles



```
<key>installer_type</key>  
<string>profile</string>
```


https://github.com/munki/munki/wiki/munki-7-removed-features

munki / munki

Type  to search

<> Code

Issues 73

Pull requests 4

Discussions

Actions

Projects

Wiki

Security

Insights

Settings

Munki 7 removed features

Greg Neagle edited this page last month · [1 revision](#)

EditNew page

The following features, supported in earlier versions of Munki, are no longer supported in Munki 7:

- The `startosinstall` `installer_type`, used to install macOS major upgrades on Intel Macs is no longer supported. The `stage_os_installer` `installer_type` works on both Intel and Apple silicon, and is supported in Munki 7.
- Installation of Apple software updates. Munki 6 does not install Apple updates on Apple silicon, but instead prompts the user to use Apple tools (System Settings app; softwareupdate) to do so. Munki 7 extends this behavior to Intel Macs as well.
- Apple update metadata (<https://github.com/munki/munki/wiki/Pkginfo-For-Apple-Software-Updates>) The `installer_type` `apple_update_metadata` is no longer supported.
- The following Adobe `installer_types` are no longer supported:
 - AdobeSetup
 - AdobeUberInstaller
 - AdobeAcrobatUpdater
 - AdobeCS5AAMEEPackage
 - AdobeCS5PatchInstaller
 - AdobeCCPInstaller

Pages 157

macOS Sequoia info

macOS Sonoma info

macOS Ventura info

Introduction

- [Getting Started](#)
- [Overview](#)
- [Discussion Group](#)
- [Demonstration Setup](#)
- [Glossary](#)
- [Frequently Asked Questions](#)
- [Contributing to Munki](#)
- [Release Notes](#)

Future Plans

Adopting Munki 7

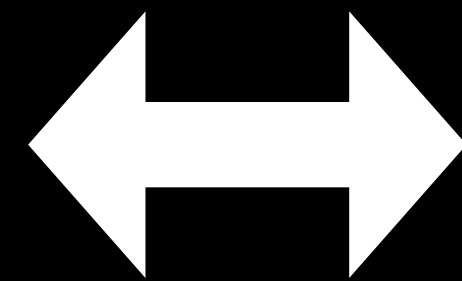
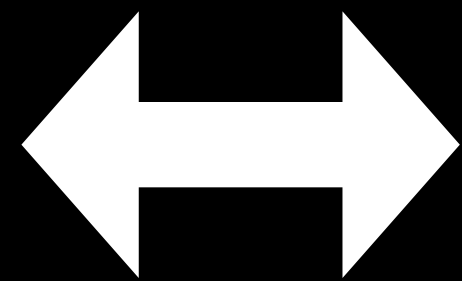
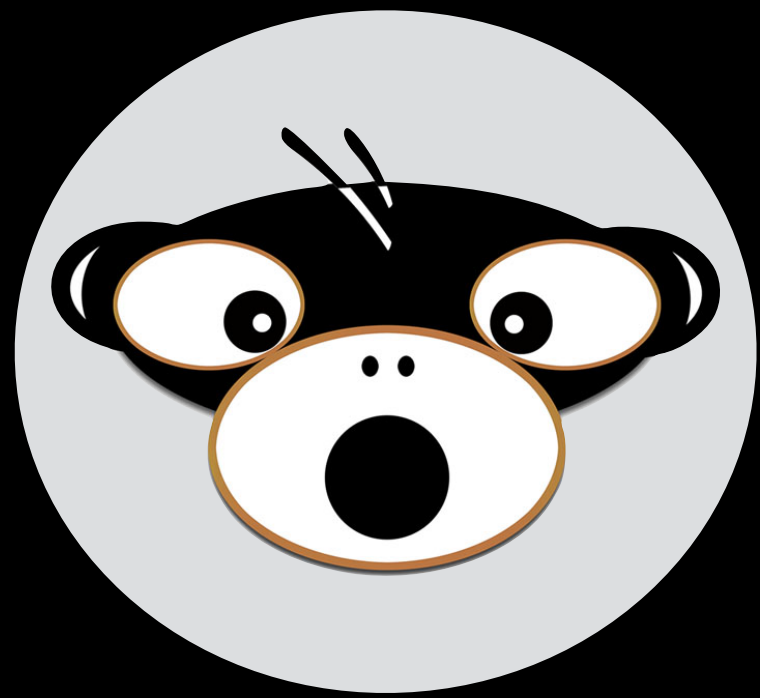
What do I need to do?



Munki 7 adoption tasks

- ☐ Deploy updated middleware
- ☐ Update TCC profile
- ☐ Deal with munki-python removal
- ☐ Audit for munkilib usage
- ☐ Check for items using removed features

Middleware



Available (Swift) middleware

<https://github.com/munki/AzureStorageMiddleware>

<https://github.com/munki/BunnyNetMiddleware>

<https://github.com/munki/CloudFrontMiddleware>

<https://github.com/munki/GCSMiddleware>

<https://github.com/munki/S3Middleware>

Privacy Preferences Policy Control

PPPC/TCC changes

Transparency, Consent, and Control



Search

Notifications

Sound

Focus

Screen Time

Lock Screen

Privacy & Security

Login Password

Users & Groups

Internet Accounts

Game Center

iCloud

Wallet & Apple Pay

Keyboard

Mouse

Printers & Scanners



Privacy & Security



Files & Folders

2 apps



Full Disk Access

2 full access



Home Accessories

None



Media & Apple Music

None



Passkeys Access for Web Browsers

None



Photos

None



Reminders

None



Accessibility

0 >



App Management

0 >



Automation

0 >



Bluetooth

0 >



Camera

0 >



Search

Notifications

Sound

Focus

Screen Time

Lock Screen

Privacy & Security

Login Password

Users & Groups

Internet Accounts

Game Center

iCloud

Wallet & Apple Pay

Keyboard

Mouse

Printers & Scanners



Privacy & Security



Files & Folders

2 apps



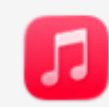
Full Disk Access

2 full access



Home Accessories

None



Media & Apple Music

None



Passkeys Access for Web Browsers

None



Photos

None



Reminders

None



Accessibility

0



App Management

0



Automation

0



Bluetooth

0



Camera

0



CodeRequirement

From:

```
<key>CodeRequirement</key>  
<string>identifier munkishim and anchor apple generic and certificate  
1[field.1.2.840.113635.100.6.2.6] /* exists */ and certificate  
leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and certificate  
leaf[subject.OU] = T4SK8ZXCXG</string>
```

To:

```
<key>CodeRequirement</key>  
<string>(identifier munkishim or identifier managedsoftwareupdate) and  
anchor apple generic and certificate 1[field.1.2.840.113635.100.6.2.6]  
/* exists */ and certificate leaf[field.1.2.840.113635.100.6.1.13] /*  
exists */ and certificate leaf[subject.OU] = T4SK8ZXCXG</string>
```


https://github.com/munki/munki/wiki/PPPC-Privacy-permissions

The screenshot shows a web browser window displaying the GitHub Wiki page for 'PPPC Privacy permissions'. The browser's address bar shows the URL 'https://github.com/munki/munki/wiki/PPPC-Privacy-permissions'. The GitHub navigation bar at the top includes the repository name 'munki / munki', a search bar, and various icons for repository management. Below the navigation bar, the page title 'PPPC Privacy permissions' is prominently displayed, followed by a note that 'Greg Neagle edited this page yesterday' and a link to '6 revisions'. The main content area contains two paragraphs: the first describes how macOS Ventura's 'App Management' privacy protection can interfere with Munki's software installation and updates; the second recommends using MDM to install a PPC/TCC profile with 'Full Disk Access' rights for Munki 6.3 and later, and provides a code snippet for the path `/usr/local/munki/managedsoftwareupdate`. A right-hand sidebar shows a 'Pages' list with 157 items, including links for 'macOS Sequoia info', 'macOS Sonoma info', 'macOS Ventura info', and an 'Introduction' section with links to 'Getting Started' and 'Overview'.

github.com

munki / munki

Type / to search

<> Code Issues 73 Pull requests 4 Discussions Actions Projects Wiki Security Insights

PPPC Privacy permissions

Greg Neagle edited this page yesterday · [6 revisions](#)

macOS Ventura added yet another new "Privacy" protection; "App Management". This feature can interfere with Munki's ability to install and update software.

With Munki 6.3 and later, it is recommended to use MDM to install a PPC/TCC profile granting either App Management or Full Disk Access rights to:

```
/usr/local/munki/managedsoftwareupdate
```

It is currently recommended to grant *Full Disk Access* rights, as that should cover almost every PPC protection third-party installers might trigger, now, and for the foreseeable future.

It's best if `managedsoftwareupdate` is signed by a Developer ID. (If it's not signed, you'll need to update your PPC/TCC profile every time there's a new release of Munki, as the `cdhash` will change.)

Pages 157

- [macOS Sequoia info](#)
- [macOS Sonoma info](#)
- [macOS Ventura info](#)
- Introduction**
 - [Getting Started](#)
 - [Overview](#)

munki-python removal



Two main strategies:

- Provide a replacement Python
- Remove/replace/update everything that depends on munki-python

Python replacements

https://github.com/gregneagle/relocatable-python

github.com

gregneagle / relocatable-python

Type / to search

+

<> Code

Issues 4

Pull requests 2

Actions

Projects

Wiki

Security

Insights

Settings

relocatable-python

Public

Pin

Unwatch 8

Fork 46

Star 160

main

1 Branch

0 Tags

Go to file

t

+

<> Code

sboiszez

add handling for paths containing symlinks (#34)

8ee72fe · 11 months ago

46 Commits

locallibs	add handling for paths containing symlinks (#...	11 months ago
.gitignore	First code commit	7 years ago
LICENSE	Initial commit	7 years ago
README.txt	Add Universal Python testing script (#26)	2 years ago
make_relocatable_python_framework...	Change shebang to /usr/bin/python3. Since th...	3 years ago
python_universal_tester.sh	Add Universal Python testing script (#26)	2 years ago
requirements_python2_recommende...	Fixes for building on Apple silicon; shebang fix...	4 years ago
requirements_python3_recommende...	Fixes for building on Apple silicon; shebang fix...	4 years ago
research_notes.txt	First code commit	7 years ago

README

Apache-2.0 license

About

A tool for building standalone relocatable Python.framework bundles

Readme

Apache-2.0 license

Activity

160 stars

8 watching

46 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Contributors

8

https://github.com/macadmins/python

macadmins / python

Type / to search

+

<> Code

Issues 3

Pull requests

Discussions

Actions

Projects

Wiki

Security

Insights

python

Public

Edit Pins

Watch 26

Fork 30

Star 248

m...

4 Branches

34 Tags

Go to file

t

+

<> Code

About

Framework files for use with popular python macadmin toolsets

Readme

erikng

Update requirements_recommended.txt

59e63bc · 3 months ago

125 Commits

.github/workflows	Add Python 3.13 ci job and update actio...	3 months ago
requirement_files	Bump requests from 2.28.1 to 2.31.0 (#...	
.gitignore	small changes	
LICENSE	Initial commit	
README.md	update all the requirement files	
build_all_python_frameworks.zsh	update gha and use new certs	
build_python_framework_pkgs....	Revert Relocatable Python hash bump (..	
example_python_script.py	Move python frameworks into their own..	
preinstall-cleanup	cleanup old scripts and move to --no-rc.	

Releases 21

Python 3.12.1.80742

on Mar 14, 2024

Latest

+ 20 releases


```
% sudo ln -s /Library/ManagedFrameworks/Python/Python3.framework/  
Versions/Current/Resources/Python.app/Contents/MacOS/Python \  
/usr/local/munki/munki-python
```



```
% sudo ln -s /usr/bin/python3 /usr/local/munki/munki-python
```


just steal the Python from Munki 6

Here's the basic idea:

- Download the Munki 6.7 installer package
- Expand the contents
- Flatten the python component package into a standalone package
- Profit!

github.com

- Apple update metadata (<https://github.com/munki/munki/wiki/Pkginfo-For-Apple-Software-Updates>) `Installer_type` `apple_update_metadata` will no longer be supported.
- `appdmg` `installer_type` (this has been deprecated for years, but support will be removed in the near future): <https://github.com/munki/munki/wiki/App-Dmg-Package-Notes>
- `suppress_bundle_relocation` `pkginfo` key. This only ever worked for bundle-style packages. Bundle-style packages are no longer supported as of macOS Sequoia, and so support for this feature will be dropped in a future release of Munki.

Contributors



chrisgrande, natewalck, and 2 other contributors

Assets3

 munkitools-6.7.0.4731.pkg	51.3 MB	last month
 Source code (zip)		last month
 Source code (tar.gz)		last month

```
% cd ~/Downloads
% curl -L https://github.com/munki/munki/releases/download/v6.7.0/munkitools-6.7.0.4731.pkg > munkitools-6.7.0.4731.pkg
```



```
% cd ~/Downloads/
```

```
% pkgutil --expand munkitools-6.7.0.4731.pkg munkitools_expanded
```

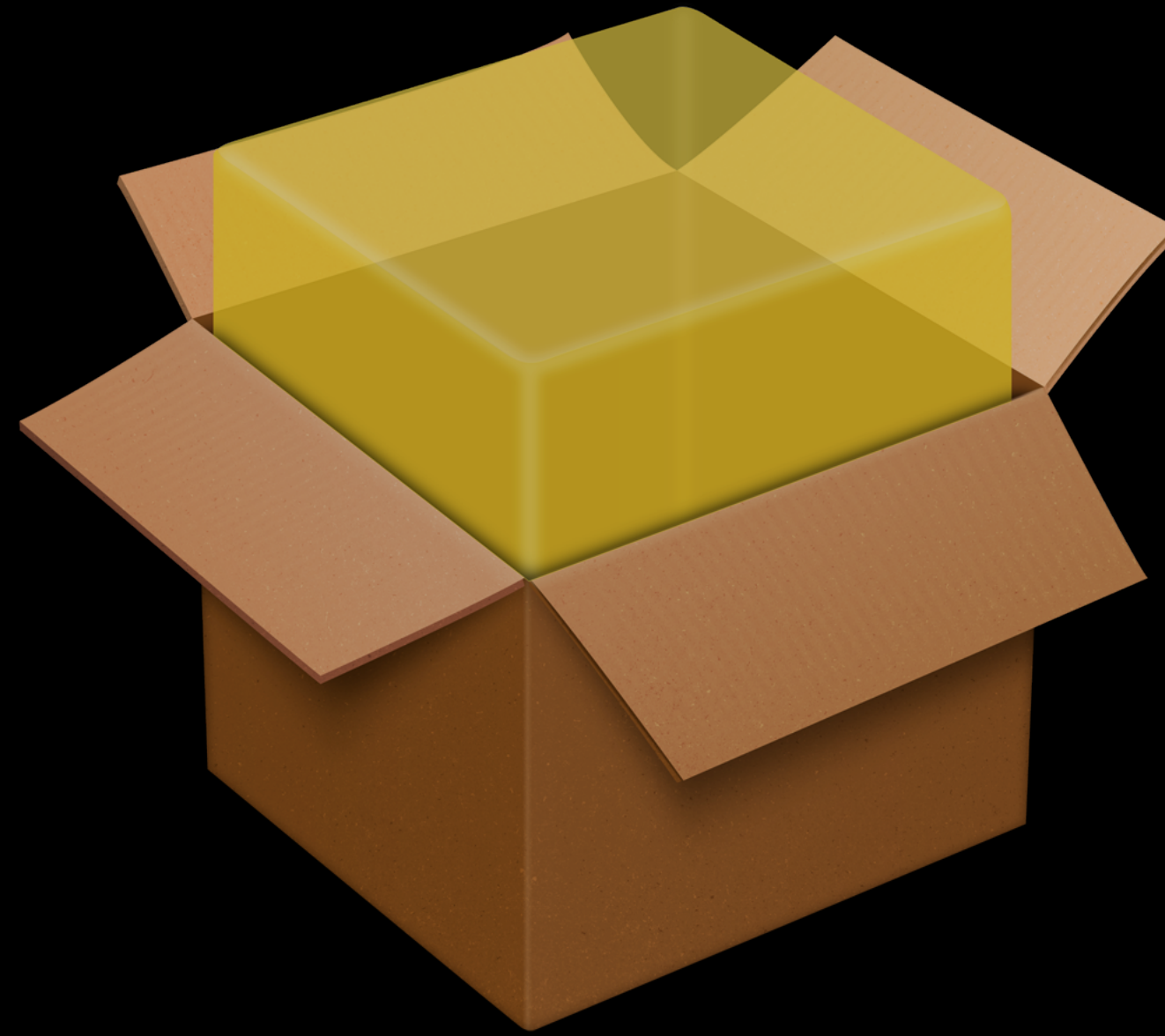


```
% ls munkitools_expanded  
Distribution  
munkitools_python.pkg  
munkitools_core.pkg.  
munkitools_launchd.pkg
```

```
munkitools_app.pkg  
munkitools_admin.pkg  
munkitools_app_usage.pkg
```



```
% pkgutil --flatten \  
munkitools_expanded/munkitools_python.pkg munkitools_python.pkg
```

```
% munkiimport munkitools_python.pkg
```


You should know:

- Python package **will not** be signed or notarized
- If you start with the signed package, the Python package you create with this method **will** install a signed Python interpreter
- munki-python will (probably) never be updated past 3.12.2



https://github.com/munki/munki/wiki/Installing-Munki-6's-Python

The screenshot shows a web browser window displaying the GitHub Munki Wiki page for 'Installing Munki 6's Python'. The browser's address bar shows the URL: `https://github.com/munki/munki/wiki/Installing-Munki-6's-Python`. The GitHub navigation bar at the top includes the Munki repository name, a search bar, and tabs for Code, Issues (73), Pull requests (4), Discussions, Actions, Projects, Wiki (selected), Security, and Insights. The page title is 'Installing Munki 6's Python', with 'Edit' and 'New page' buttons to the right. Below the title, it says 'Greg Neagle edited this page 5 minutes ago · [3 revisions](#)'. The main content area has an 'Introduction' section with two paragraphs and a 'Steps' section with two numbered items. A code block at the bottom shows terminal commands. On the right, a sidebar contains a 'Pages' list with 160 items, including links for macOS Tahoe, Sequoia, and Sonoma info, and a list of other wiki pages like 'Getting Started' and 'Release Notes'.

`https://github.com/munki/munki/wiki/Installing-Munki-6's-Python`

munki / munki

Type `/` to search

Code Issues 73 Pull requests 4 Discussions Actions Projects Wiki Security Insights

Installing Munki 6's Python

Greg Neagle edited this page 5 minutes ago · [3 revisions](#)

[Edit](#) [New page](#)

Introduction

Munki 7 no longer includes a Python interpreter. If you have scripts or other tools that currently depend on Munki's Python, and you have not yet had time to update them or replace them with alternatives that do not depend on Munki's Python, you might be looking for a way to continue to use Munki 6's Python for a limited time as you address these dependencies.

Clients that are upgraded from Munki 6 to Munki 7 will *not* have Munki's Python removed, but machines that have never had a previous version of Munki installed won't have the Python from Munki unless you do extra work.

Here is a method to create an Apple package that installs Munki 6's Python.

Steps

1. Download the installer package for a recent release of Munki 6. The following steps are for the Munki 6.7 release.
2. Assuming the download is in `~/Downloads`, at the command line:

```
% cd ~/Downloads
% pkgutil --expand munkitools-6.7.0.4731.pkg munkitools_expanded
```

[Pages](#) 160

[macOS Tahoe info](#)

[macOS Sequoia info](#)

[macOS Sonoma info](#)

Introduction

- [Getting Started](#)
- [Overview](#)
- [Discussion Group](#)
- [Demonstration Setup](#)
- [Glossary](#)
- [Frequently Asked Questions](#)
- [Contributing to Munki](#)
- [Release Notes](#)

Future Plans

- [Deprecation Notes](#)

Updating tools with dependencies on Munki's Python

A photograph of a small, shallow stream flowing over a bed of smooth, grey and white rocks. The water is clear and reflects the surrounding green foliage. The stream is bordered by dense vegetation, including tall grasses and leafy plants, some of which are slightly out of focus. The overall scene is a natural, outdoor setting with soft lighting.

/usr/local/munki/conditions/


```
% grep -R munki-python /usr/local/munki/conditions/  
/usr/local/munki/conditions/munki_facts.py:  
    #!/usr/local/munki/munki-python
```


https://github.com/munki/munki-facts

github.com

munki / **munki-facts**

Type / to search

<> Code

Issues

Pull requests 1

Actions

Projects

Wiki

Security

Insights

Settings

munki-facts

Public

Edit Pins

Unwatch 16

Fork 30

Star 50

master

2 Branches

0 Tags

Go to file

+

<> Code

gregneagle

Merge pull request #36 from JiiJuuTee/patch-1

a22a02a · 2 months ago

88 Commits

facts	Add supported models for macOS 26 Tahoe	2 months ago
.gitignore	Initial commit	9 years ago
LICENSE.md	Create LICENSE.md	8 years ago
README.md	Update README.md	8 years ago
munki_facts.py	Fix for Python 3.10.11+	2 years ago

README

License

Introduction

munki_facts.py is an "admin-provided conditions" script for Munki as described [here](#).

munki_facts.py is designed to be modular, allowing more 'facts' to be easily added without duplicating all of the code required to insert the facts into the ConditionalItems.plist.

About

A framework for "admin-provided conditionals" for Munki.

Readme

View license

Activity

Custom properties

50 stars

16 watching

30 forks

Report repository

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

https://github.com/MagerValp/bash-facts

https://github.com/MagerValp/bash-facts

MagerValp / bash-facts

Type / to search

<> Code

Pull requests

Actions

Projects

Security

Insights

bash-facts

Public

forked from [munki/munki-facts](#)

Watch

2

Fork

0

Star

6

ma...

3 Branches

0 Tags

Go to file

t

+

<> Code

This branch is 6 commits ahead of, 4 commits behind [munki/munki-facts:master](#).

MagerValp

Add console_user.sh

577d1a8 · 3 months ago

🕒 90 Commits

📁 conditions	Add console_user.sh	3 months ago
📄 .gitignore	Initial commit	9 years ago
📄 LICENSE.md	Create LICENSE.md	8 years ago
📄 README.md	Update README.md	3 months ago

Readme

View license

Activity

6 stars

2 watching

0 forks

Report repository

scripts in pkginfo items

```
% cd /path/to/munki_repo  
% grep -R munki-python pkginfo
```

```
pkginfo/apps/Adobe/AdobeReaderDC-25.001.20630.plist:  
    <string>#!/usr/local/munki/munki-python  
pkginfo/apps/Adobe/AdobeReaderDC-25.001.20693.plist:  
    <string>#!/usr/local/munki/munki-python
```




apps/Adobe/AdobeReaderDC-25.001.20693.plist

[Basics](#)[Details](#)[plist](#)

installcheck_script

```
#!/usr/local/munki/munki-python
```

```
import os
import plistlib
import sys
sys.path.insert(0, '/usr/local/munki')
from munkilib.pkgutils import MunkiLooseVersion as LooseVersion

def read_plist(plist_path):
    """Reads a plist file and returns its contents as a
    dictionary."""
    print(f"Reading plist from: {plist_path}")
    if os.path.exists(plist_path):
        with open(plist_path, 'rb') as plist_file:
            return plistlib.load(plist_file)
    else:
        print(f"Plist not found: {plist_path}")
        return None
```


Update AdobeReader.munki.recipe (#505)

* Update AdobeReader.munki.recipe

Refactors the `installcheck\_script` from `munki-python` to `zsh` for future compatibility with Munki 7

* Update AdobeReader.munki.recipe

D'oh! - removes the hardcoded version number

 [master](#) (#505)

1 parent [a2f4aa3](#) commit 5049050 

 Filter files...



✓  AdobeReader

 AdobeReader.munki.recipe



1 file changed +69 -85 lines changed

 Search within code



▼ AdobeReader/AdobeReader.munki.recipe  

+69 -85     ...

 @@ -143,105 +143,89 @@ exit 0

143

`<key>version</key>`

144

`<string>%app_version%</string>`

145

`<key>installcheck_script</key>`

146

-

`<string>#!/usr/local/munki/munki-
python`

143

`<key>version</key>`

144

`<string>%app_version%</string>`

145

`<key>installcheck_script</key>`

146

+

`<string>#!/bin/zsh`

147

+ #

148

+ # DESCRIPTION

149

+ # - Exits with code 1 if Adobe
Acrobat DC Unified App is installed

Update AdobeReader.munki.recipe (#505)

* Update AdobeReader.munki.recipe

Refactors the `installcheck\_script` from `munki-python` to `zsh` for future compatibility with Munki 7

* Update AdobeReader.munki.recipe

D'oh! - removes the hardcoded version number

 [master](#) (#505)

1 parent [a2f4aa3](#) commit 5049050 

 Filter files...



✓  AdobeReader

 AdobeReader.munki.recipe

 **1 file changed** +69 -85 lines changed

 Search within code



▼ AdobeReader/AdobeReader.munki.recipe  

+69 -85      ...

 @@ -143,105 +143,89 @@ exit 0

```
143     <key>version</key>
144     <string>%app_version%</string>
145     <key>installcheck_script</key>
146 -    <string>#!/usr/local/munki/munki-
python
```

```
143     <key>version</key>
144     <string>%app_version%</string>
145     <key>installcheck_script</key>
```

```
146 +    <string>#!/bin/zsh
147 + #
148 + # DESCRIPTION
149 + # - Exits with code 1 if Adobe
Acrobat DC Unified App is installed
```


munki-python: some more places to look

```
/usr/local/munki/conditions/  
/usr/local/munki/postflight  
/usr/local/munki/postflight.d/  
/usr/local/munki/preflight  
/usr/local/munki/preflight.d/
```

```
/Library/ManagementScripts/ (or whatever)  
/Library/YourOrganization/ (or whatever)
```


munkilib dependencies


```
% ls /usr/local/munki/munkilib/
```

__init__.py	FoundationPlist.py	osutils.py
__pycache__	gurl.py	pkgutils.py
admin	iconutils.py	powermgr.py
adobeutils	info.py	prefs.py
app_usage.py	installer	processes.py
appleupdates	installinfo.py	profiles
authrestart	keychain.py	removepackages.py
bootstrapping.py	launchd	reports.py
cliutils.py	munkicommon.py	scriptutils.py
constants.py	munkihash.py	svnversion
dateutils.py	munkilog.py	updatecheck
display.py	munkirepo	utils.py
dmgutils.py	munkistatus.py	version.plist
fetch.py	osinstaller.py	wrappers.py


```
#!/usr/local/munki-python
```

```
from munkilib import appleupdates  
from munkilib import authrestart  
from munkilib import bootstrapping  
from munkilib import constants  
from munkilib import dateutils  
from munkilib import display  
from munkilib import info  
from munkilib import installer  
from munkilib import installinfo  
from munkilib import munkilog  
from munkilib import munkistatus  
from munkilib import osinstaller  
from munkilib import osutils  
from munkilib import powermgr  
from munkilib import prefs  
from munkilib import processes
```


github.com

autopkg/Code/autopkglib/munkirepolibs/MunkiLib.py at master · autopkg/autopkg

master

autopkg / Code / autopkglib / munkirepolibs / MunkiLib.py

↑ Top

Code

Blame

Raw

```
5  from autopkglib import ProcessorError
6
7
8  class MunkiLib:
9      def __init__(self, munki_repo, munki_repo_plugin, munkilib_dir, repo_subdirectory):
10         self.repo_subdirectory = repo_subdirectory
11         self.munki_repo = munki_repo
12
13         sys.path.insert(0, munkilib_dir)
14         try:
15             from munkilib import munkirepo
16             from munkilib.admin import munkiimportlib
17             from munkilib.cliutils import path2url
18         except ImportError as err:
19             raise ProcessorError(
20                 f"munkilib import error: {str(err)}\nMunki tools version 3.2.0.3462 or "
21                 "later is required."
22             )
23
```

Symbols

Find definitions and references for other symbols in this file below or in the code.

Filter symbols

class

 MunkiLib

func

 __init__

func

 _full_path

func

 make_cat

func

 copy_pkg

func

 copy_pkg

func

 find_match

func

 extract_artifact

AutoPkg

github.com

sal/server/plugins/munkiinfo/scripts/munkiinfo.py at main · salopensource/sal

main

sal / server / plugins / munkiinfo / scripts / munkiinfo.py

↑ Top

Code

Blame

Raw

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

import sys

import sal

sys.path.append('/usr/local/munki')

from munkilib import munkicommon

PREFS_TO_GET = (

'ManagedInstallDir',

'SoftwareRepoURL',

'ClientIdentifier',

'LogFile',

'LoggingLevel',

'LogToSyslog',

'InstallAppleSoftwareUpdates',

'AppleSoftwareUpdatesOnly',

'SoftwareUpdateServerURL'

)

Symbols

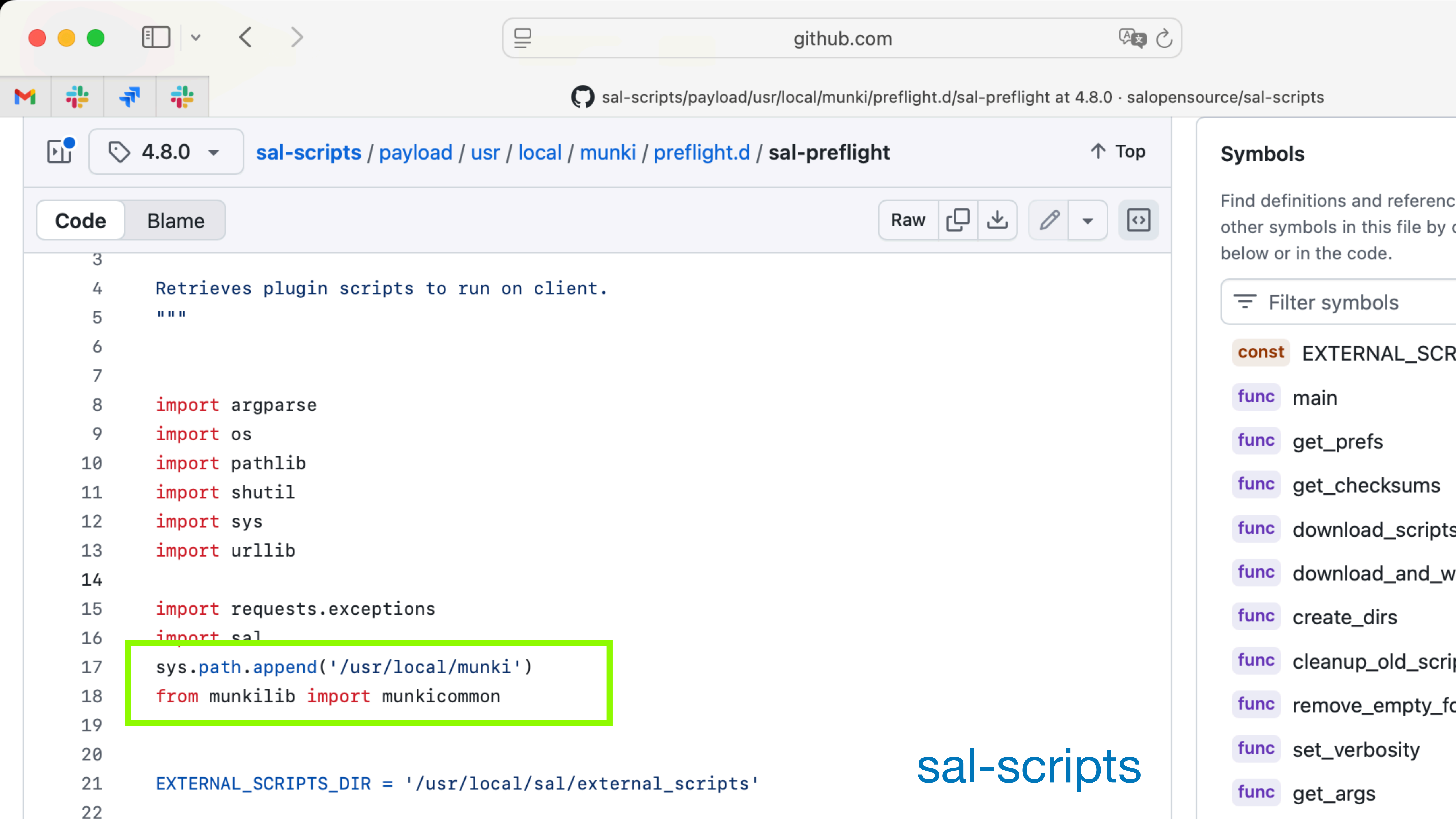
Find definition
other symbol
below or in th

Filter s

const PR

func main

Sal



github.com

sal-scripts/payload/usr/local/munki/preflight.d/sal-preflight at 4.8.0 · salopensource/sal-scripts



4.8.0

sal-scripts / payload / usr / local / munki / preflight.d / sal-preflight

↑ Top

Code

Blame

Raw



```
3
4     Retrieves plugin scripts to run on client.
5     """
6
7
8     import argparse
9     import os
10    import pathlib
11    import shutil
12    import sys
13    import urllib
14
15    import requests.exceptions
16    import sal
17    sys.path.append('/usr/local/munki')
18    from munkilib import munkicommon
19
20
21    EXTERNAL_SCRIPTS_DIR = '/usr/local/sal/external_scripts'
22
```

Symbols

Find definitions and references for other symbols in this file by clicking on the symbol name below or in the code.

Filter symbols

const EXTERNAL_SCR

func main

func get_prefs

func get_checksums

func download_scripts

func download_and_w

func create_dirs

func cleanup_old_scrip

func remove_empty_fc

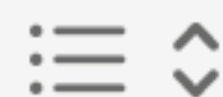
func set_verbosity

func get_args

sal-scripts



munkilib



Name



Date Modified

Size

Kind



__init__.py

Sep 19, 2025 at 1:22 PM

144 bytes

Python Source



> __pycache__

Sep 19, 2025 at 1:34 PM

--

Folder



> admin

Sep 19, 2025 at 1:33 PM

--

Folder



> adobeutils

Sep 19, 2025 at 1:33 PM

--

Folder



app_usage.py

Sep 19, 2025 at 1:22 PM

18 KB

Python Source



> appleupdates

Sep 19, 2025 at 1:33 PM

--

Folder



> authrestart

Sep 19, 2025 at 1:33 PM

--

Folder



bootstrapping.py

Sep 19, 2025 at 1:22 PM

5 KB

Python Source



cliutils.py

Sep 19, 2025 at 1:22 PM

9 KB

Python Source



constants.py

Sep 19, 2025 at 1:22 PM

2 KB

Python Source



dateutils.py

Sep 19, 2025 at 1:22 PM

3 KB

Python Source



display.py

Sep 19, 2025 at 1:22 PM

7 KB

Python Source



dmgutils.py

Sep 19, 2025 at 1:22 PM

9 KB

Python Source



fetch.py

Sep 19, 2025 at 1:22 PM

26 KB

Python Source



FoundationPlist.py

Sep 19, 2025 at 1:22 PM

5 KB

Python Source



gurl.py

Sep 19, 2025 at 1:22 PM

37 KB

Python Source



iconutils.py

Sep 19, 2025 at 1:22 PM

11 KB

Python Source



info.py

Sep 19, 2025 at 1:22 PM

34 KB

Python Source



Macintosh HD >



usr >



local >



munki >



munkilib

Munki 7 adoption tasks

- ☒ Deploy updated middleware
- ☒ Update TCC profile
- ☒ Deal with munki-python removal
- ☒ Audit for munkilib usage
- ☐ Check for items using removed features

github.com

Munki 7 removed features · munki/munki Wiki

munki / munki

Type / to search

<> Code

Issues 73

Pull requests 4

Discussions

Actions

Projects

Wiki

Security

Insights

Munki 7 removed features

Greg Neagle edited this page 2 days ago · [3 revisions](#)

Edit

New page

The following features, supported in earlier versions of Munki, are no longer supported in Munki 7:

- The `startosinstall` `installer_type`, used to install macOS major upgrades on Intel Macs is no longer supported. The `stage_os_installer` `installer_type` works on both Intel and Apple silicon, and is supported in Munki 7.
- Installation of Apple software updates. Munki 6 does not install Apple updates on Apple silicon, but instead prompts the user to use Apple tools (System Settings app; softwareupdate) to do so. Munki 7 extends this behavior to Intel Macs as well.
- Apple update metadata (<https://github.com/munki/munki/wiki/Pkginfo-For-Apple-Software-Updates>)
The `installer_type` `apple_update_metadata` is no longer supported.
- The following Adobe `installer_types` are no longer supported:
 - AdobeSetup
 - AdobeUberInstaller
 - AdobeAcrobatUpdater
 - AdobeCS5AAMEEPackage
 - AdobeCS5PatchInstaller
 - AdobeCCPIInstaller

Pages 160

macOS Tahoe info

macOS Sequoia info

macOS Sonoma info

Introduction

- [Getting Started](#)
- [Overview](#)
- [Discussion Group](#)
- [Demonstration Setup](#)
- [Glossary](#)
- [Frequently Asked Questions](#)
- [Contributing to Munki](#)
- [Release Notes](#)

Future Plans


```
1 //
2 //  main.swift
3 //  repocheck
4 //
5 //  Created by Greg Neagle on 9/30/25.
6 //
7
8 import Foundation
9
10 func getPkginfoList(repo: Repo) async throws -> [String] {
11     do {
12         return try await repo.list("pkginfo")
13     } catch is MunkiError {
14         throw MunkiError("Error getting list of pkginfo items")
15     }
16 }
17
18 func installerTypeIsSupportedInMunki7(_ installerType: String) -> Bool {
19     if ["appdmg", "apple_update_metadata", "startosinstall",
20         "profile"].contains(installerType) {
21         return false
22     }
23     if installerType.hasPrefix("Adobe") {
24         return false
25     }
26     return true
27 }
28
29 func uninstallMethodIsSupportedInMunki7(_ uninstallMethod: String) -> Bool {
30     return !uninstallMethod.hasPrefix("Adobe")
31 }
```



```
% repocheck
os_updates/InstallBigSur-11.7.10-x86_64.plist:
    Unsupported installer_type: startosinstall

os_updates/InstallMonterey-12.7.6-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallBigSur-11.7.9.plist:
    Unsupported installer_type: startosinstall

os_updates/InstallMonterey-12.7.5-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallCatalina-10.15.7.15.plist:
    Unsupported installer_type: startosinstall

os_updates/InstallVentura-13.7.8-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallSequoia-15.6.1-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallSonoma-14.8-x86_64:
    Unsupported installer_type: startosinstall
```



```
% repocheck
os_updates/InstallBigSur-11.7.10-x86_64.plist:
    Unsupported installer_type: startosinstall

os_updates/InstallMonterey-12.7.6-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallBigSur-11.7.9.plist:
    Unsupported installer_type: startosinstall

os_updates/InstallMonterey-12.7.5-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallCatalina-10.15.7.15.plist:
    Unsupported installer_type: startosinstall

os_updates/InstallVentura-13.7.8-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallSequoia-15.6.1-x86_64:
    Unsupported installer_type: startosinstall

os_updates/InstallSonoma-14.8-x86_64:
    Unsupported installer_type: startosinstall
```


apps/Adobe/CC2019/AdobeLightroomClassicCC2019-8.4.1.201909111555.plist:

Unsupported installer_type: AdobeCCPInstaller

Unsupported uninstall_method: AdobeCCPUninstaller

apps/Adobe/CC2019/AdobeMediaEncoderCC2019-13.1.5.35.plist:

Unsupported installer_type: AdobeCCPInstaller

Unsupported uninstall_method: AdobeCCPUninstaller

apps/Adobe/CC2019/AdobePhotoshopCC2019-20.0.10.plist:

Unsupported installer_type: AdobeCCPInstaller

Unsupported uninstall_method: AdobeCCPUninstaller

apps/Adobe/CC2019/AdobeAnimateCC2019-19.2.1.408.plist:

Unsupported installer_type: AdobeCCPInstaller

Unsupported uninstall_method: AdobeCCPUninstaller

apps/Adobe/CC2019/AdobeIllustratorCC2019-23.1.1.999.plist:

Unsupported installer_type: AdobeCCPInstaller

Unsupported uninstall_method: AdobeCCPUninstaller

100/2977 pkginfos have issues

Munki 7

- ☒ What's new
- ☒ What's gone
- ☒ How to get ready

when Munki 7?



- 15 betas starting late May 2025
- 3 Release Candidates over the last week

Right now

A photograph of a rugged, rocky mountain landscape under a clear blue sky. The foreground shows a dirt path leading towards a steep, rocky slope. The word "Tack!" is overlaid in white text in the center of the image.

Tack!