

Writing AutoPkg Recipes

A Workshop for EveryWorld

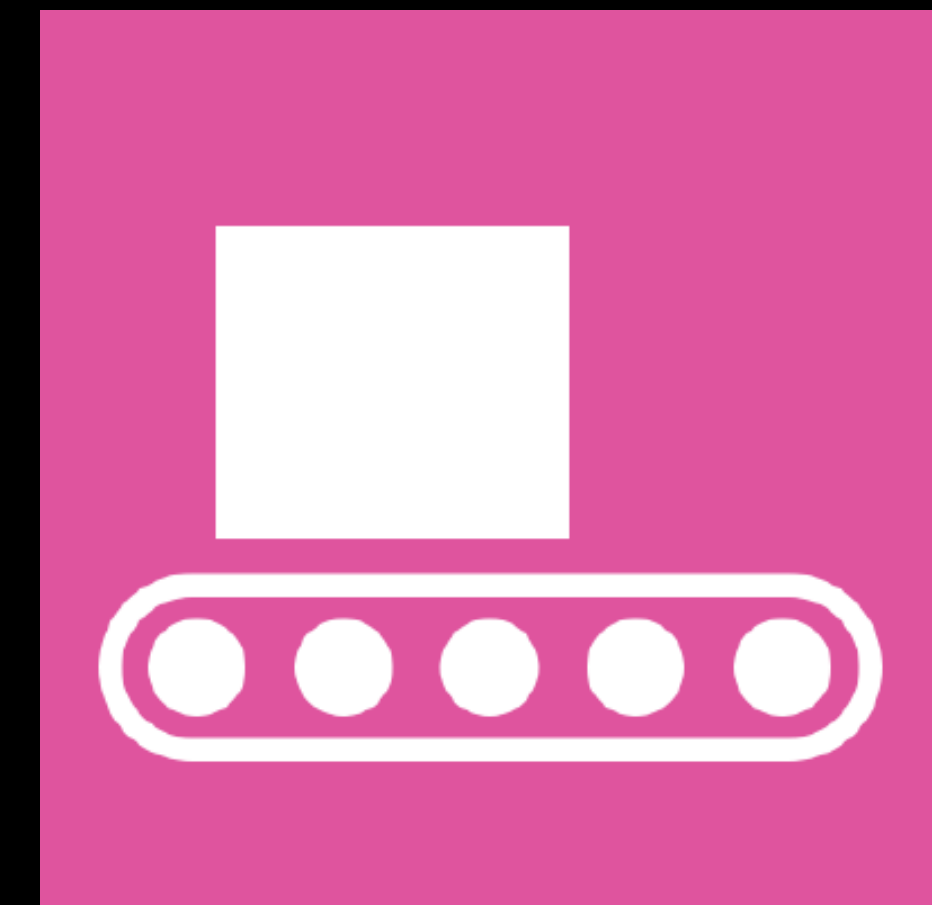
Anthony Reimer

 maclabs.jazzace.ca

 [AnthonyReimer](https://twitter.com/AnthonyReimer)

 [jazzace](https://jazzace.ca)

 [jazzace](https://github.com/jazzace)



**UNIVERSITY OF
CALGARY**

Writing AutoPkg Recipes

A Workshop

Anthony Reimer

 maclabs.jazzace.ca

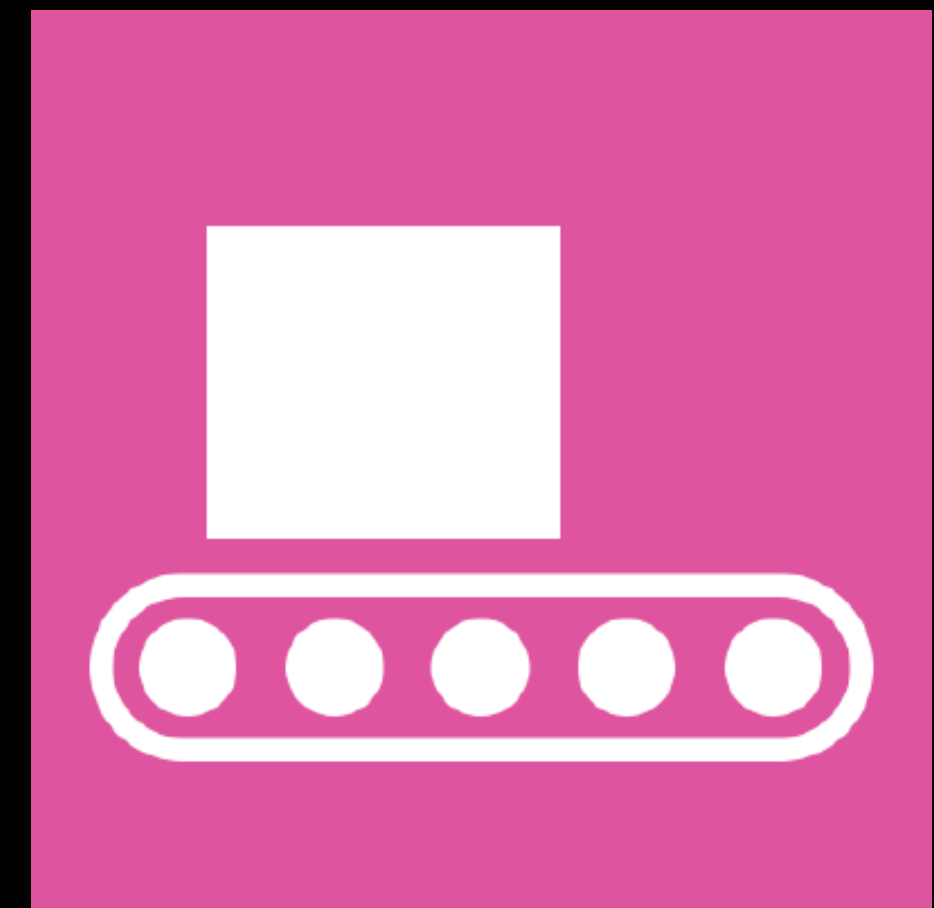
 [AnthonyReimer](https://twitter.com/AnthonyReimer)

 [jazzace](http://jazzace.com)

 [jazzace](https://github.com/jazzace)



Elliot Jordan



**UNIVERSITY OF
CALGARY**

Workshop Outline

Prerequisites

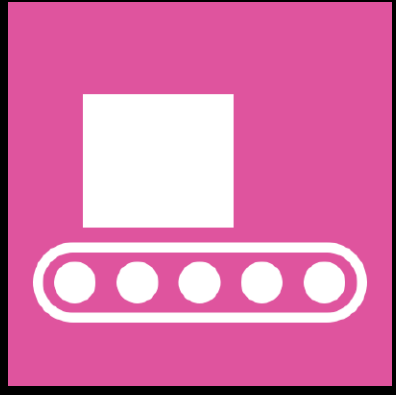
Recipe Reading & Literacy

Creating a new recipe

Testing your recipe

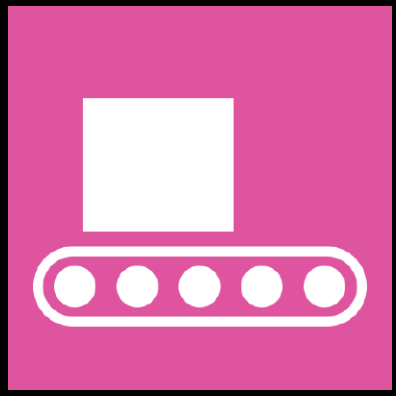
Writing a download recipe LIVE!





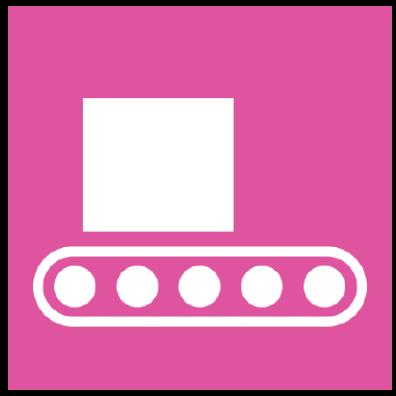
What You Need

- AutoPkg 2.2 with core recipes repo
 - ▶ *github.com/autopkg/autopkg/releases*
 - ▶ `autopkg repo-add recipes`
- Git
 - ▶ *Xcode Command Line Tools or git-scm.com/download/mac*
- Plain Text Editor (e.g., BBEdit)
- Terminal



Optional Installs

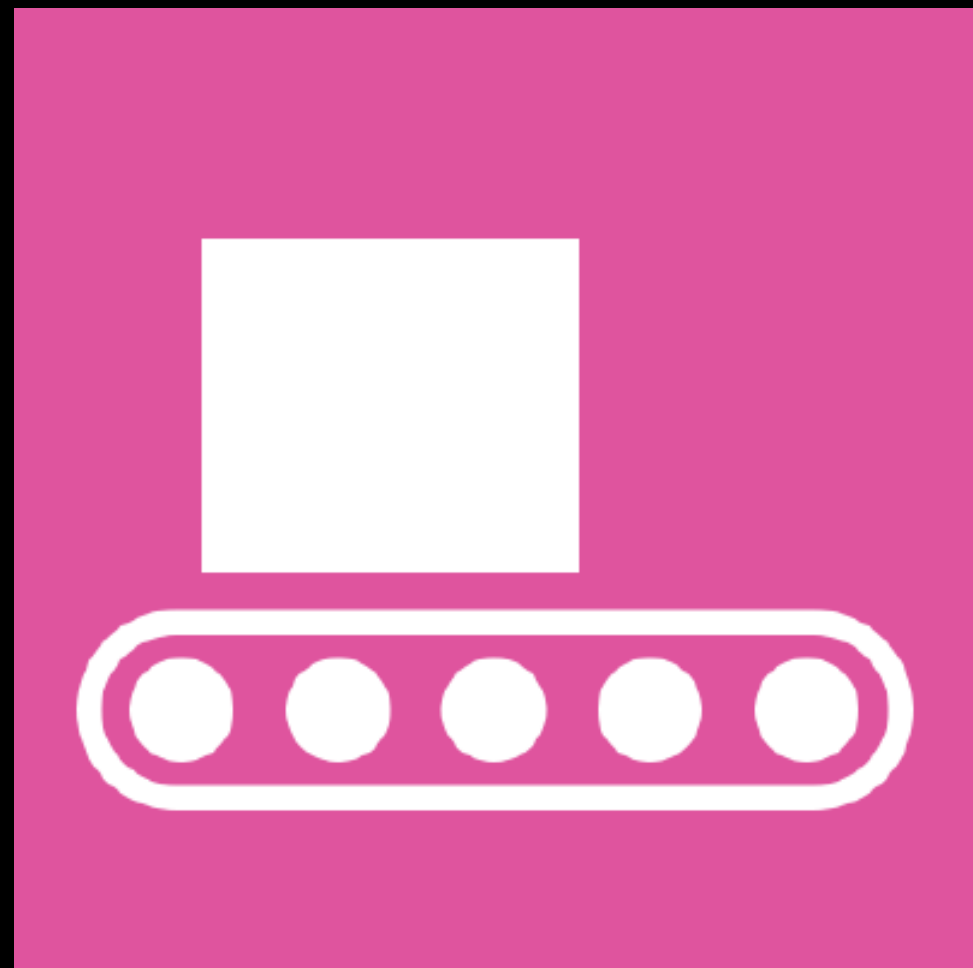
- **AutoPkgr 1.5.5**
 - ▶ *github.com/lindegrouper/autopkgr/releases*
- **Recipe Robot 2**
 - ▶ *github.com/homebysix/recipe-robot/releases*
- **RecipeBuilder**
 - ▶ *github.com/mikaellofgren/RecipeBuilder/releases*

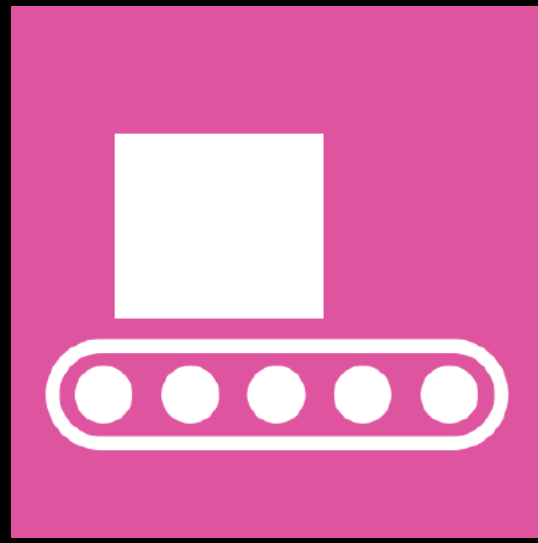


Basic User Workflow

1. Search for recipes
2. Add related repo(s)
3. Create overrides, editing Input variables if needed
4. Run overrides
5. Maintain: Update repos, update Override trust information

AutoPkg v. AutoPkggr





command line



User interface

Runs AutoPkg

Assisted Initial Setup

Access to all run options

Processor list & info

Includes Scheduled Runs

Includes Notifications

Supports Trust verification



GUI



Recipe Reading & Literacy

Self Saucing Fudge Dessert

When Joan serves this to company, she puts a scoop of ice cream on the pudding base before topping it with the sauce.

pudding Base

1 cup	flour	250 mL
2 tsp.	baking powder	10 mL
¼ tsp.	salt	1 mL
½ cup	sugar	125 mL
3 tbsp.	cocoa	50 mL
½ cup	chopped nuts	125 mL
½ cup	milk	125 mL
1 tsp.	vanilla	5 mL
2 tbsp.	melted butter or margarine	25 mL

Sauce

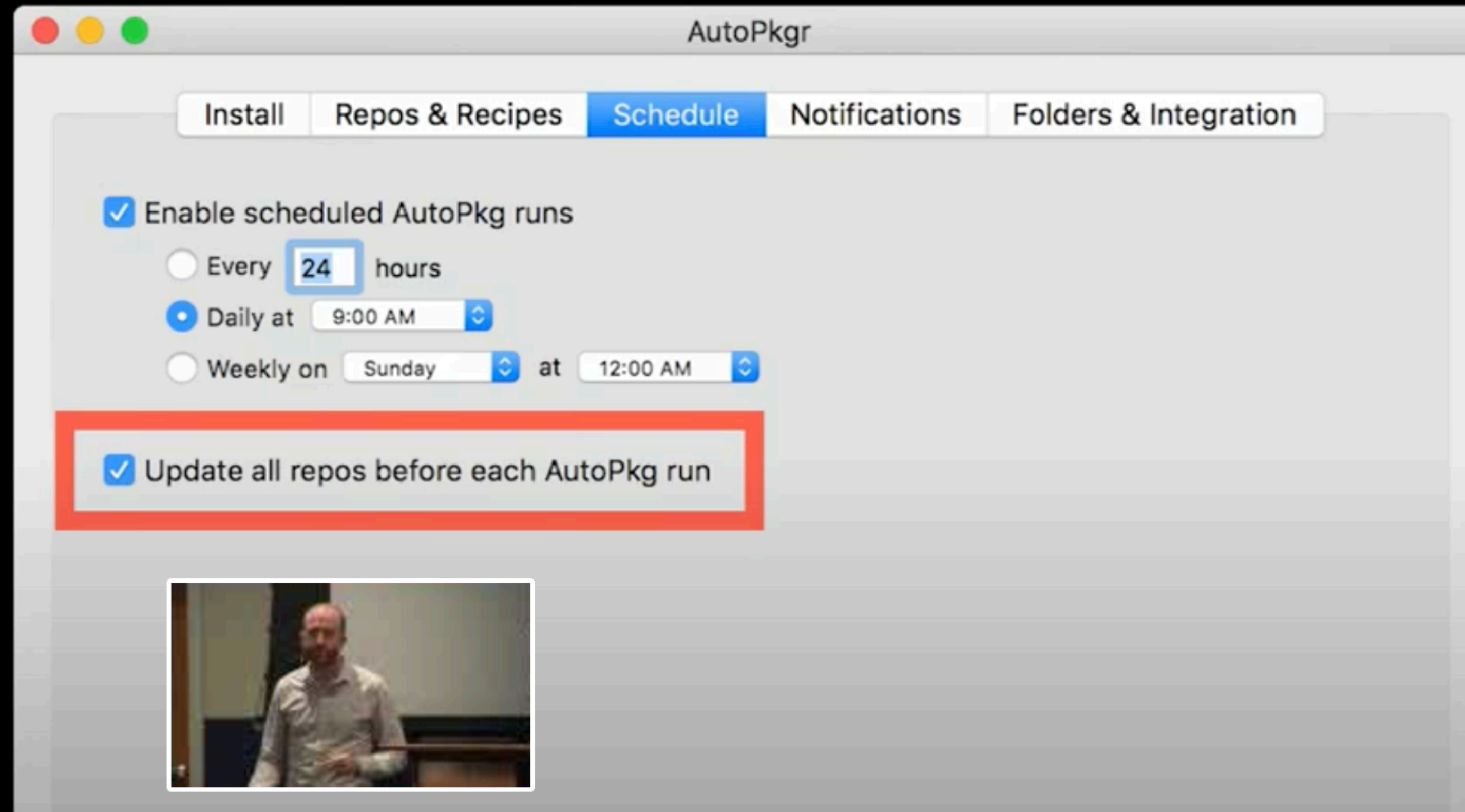
4 tbsp.	cocoa	65 mL
½ cup	sugar	125 mL
2 cups	hot water	500 mL

Sift into a two quart (litre) casserole, flour, baking powder, salt, ½ cup (125 mL) of sugar and 3 tablespoons (50 mL) of cocoa. Stir in the nuts. Make a well in the centre and add milk, vanilla and melted butter. Stir lightly and smooth evenly over bottom of the casserole.

Mix ½ cup (125 mL) of sugar and 4 tablespoons (65 mL) of cocoa. Sprinkle mixture over pudding base. Pour hot water over cocoa sugar mixture. Bake at 350°F (180°C) for 30 minutes. The batter rises and a rich sauce forms under it. Serve hot.



#7: Trust Without Verification



How (Not) To Do Bad Things With AutoPkg – Elliot Jordan
<https://youtube.com/watch?v=LnvQmLcDF8w>

Name



Description



Ingredients



Instructions



Self Saucing Fudge Dessert

When Joan serves this to company, she puts a scoop of ice cream on the pudding base before topping it with the sauce.

Pudding Base

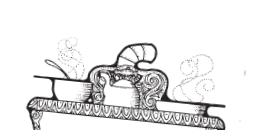
1 cup	flour	
2 tsp.	baking powder	250 mL
¼ tsp.	salt	10 mL
½ cup	sugar	1 mL
3 tbsp.	cocoa	125 mL
½ cup	chopped nuts	50 mL
½ cup	milk	125 mL
1 tsp.	vanilla	125 mL
2 tbsp.	melted butter or margarine	5 mL
		25 mL

Sauce

4 tbsp.	cocoa	
½ cup	sugar	65 mL
2 cups	hot water	125 mL
		500 mL

Sift into a two quart (litre) casserole, flour, baking powder, salt, ½ cup (125 mL) of sugar and 3 tablespoons (50 mL) of cocoa. Stir in the nuts. Make a well in the centre and add milk, vanilla and melted butter. Stir lightly and smooth evenly over bottom of the casserole.

Mix ½ cup (125 mL) of sugar and 4 tablespoons (65 mL) of cocoa. Sprinkle mixture over pudding base. Pour hot water over cocoa sugar mixture. Bake at 350°F (180°C) for 30 minutes. The batter rises and a rich sauce forms under it. Serve hot.

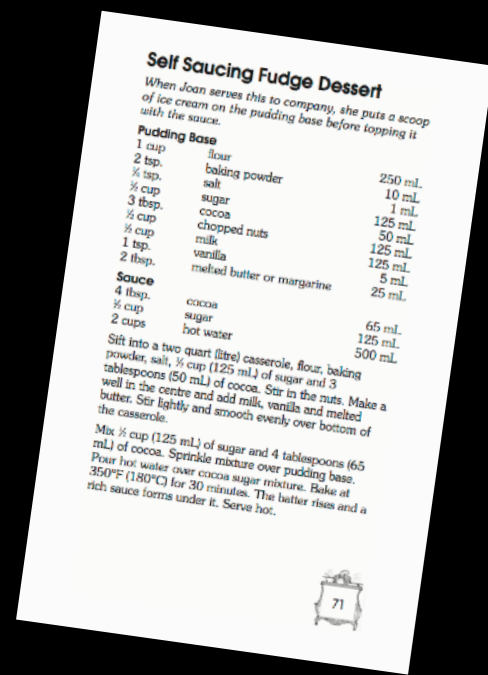


Identifier

Description

Input

Process



About AutoPkg Recipes

- Plist
- Series of steps (Processors)
- Can leverage other recipes ("Parent-Child")
- Best practice is to create a recipe chain rather than an omnibus (monolithic) recipe

Nouns & Verbs

.download

.munki

.pkg

.jss

.filewave

.install

Adjectives

Parent

Child

Override

Stub

XML

YAML

.download

Parent

.munki

.pkg

Override

Child

They're all just *recipes*

.jss

Stub

.install

XML

.filewave

YAML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
```

Doc Type Def

```
<dict>
```

```
<key>Description</key>
<string>This recipe does nothing.</string>
```

Description

```
<key>Identifier</key>
<string>com.github.yourgithubusername.download.nothing</string>
```

Identifier

```
<key>Input</key>
<dict>
  <key>NAME</key>
  <string>nothing</string>
</dict>
```

Input

```
<key>MiniumumVersion</key>
<string>1.0</string>
```

Minimum Version

```
<key>Process</key>
<array>
  <dict>
    <key>Arguments</key>
    <dict>
      <key>Argument1</key>
      <string>Value1</string>
      <key>Argument2</key>
      <string>Value2</string>
    </dict>
    <key>Processor</key>
    <string>EndOfCheckPhase</string>
  </dict>
</array>
```

Processor(s)

Process

```
</dict>
```

```
</plist>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
```

```
<key>Description</key>
<string>This recipe calls a recipe that does nothing.</string>
```

Description

```
<key>Identifier</key>
<string>com.github.yourgithubusername.pkg.nothing</string>
```

Identifier

```
<key>Input</key>
<dict>
  <key>NAME</key>
  <string>nothing</string>
</dict>
```

Input

```
<key>MiniumumVersion</key>
<string>1.0</string>
```

MinimumVersion

```
<key>ParentRecipe</key>
<string>com.github.yourgithubusername.download.nothing</string>
```

ParentRecipe

```
<key>Process</key>
<array>
  <dict>
    <key>Arguments</key>
    <dict>
      <key>Argument1</key>
      <string>Value1</string>
    </dict>
    <key>Processor</key>
    <string>AppPkgCreator</string>
  </dict>
</array>
```

Processor(s)

Process

```
</dict>
</plist>
```



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
<key>Identifier</key>
<string>local.pkg.nothing</string>
<key>Input</key>
<dict>
<key>NAME</key>
<string>nothing</string>
</dict>
<key>ParentRecipe</key>
<string>com.github.yourgithubusername.pkg.nothing</string>
<key>ParentRecipeTrustInfo</key>
<dict>
<key>non_core_processors</key>
<dict/>
<key>parent_recipes</key>
<dict>
<key>com.github.yourgithubusername.download.nothing</key>
<dict>
<key>path</key>
<string>~/Library/AutoPkg/Recipes/nothing.download.recipe</string>
<key>sha256_hash</key>
<string>fda53c4c7dbb1550aa9a888af14537bd42f16d73bb48da685b4da70fcc2e8500</string>
</dict>
<key>com.github.yourgithubusername.pkg.nothing</key>
<dict>
<key>path</key>
<string>~/Library/AutoPkg/Recipes/nothing.pkg.recipe</string>
<key>sha256_hash</key>
<string>463565c8aaf8c02465b611c589d3cb83a602c288b184f3e0b49d8d435bb10406</string>
</dict>
</dict>
</dict>
</plist>
```

Identifier

Input

ParentRecipe

non_core_processors

ParentRecipeTrustInfo

parent_recipes

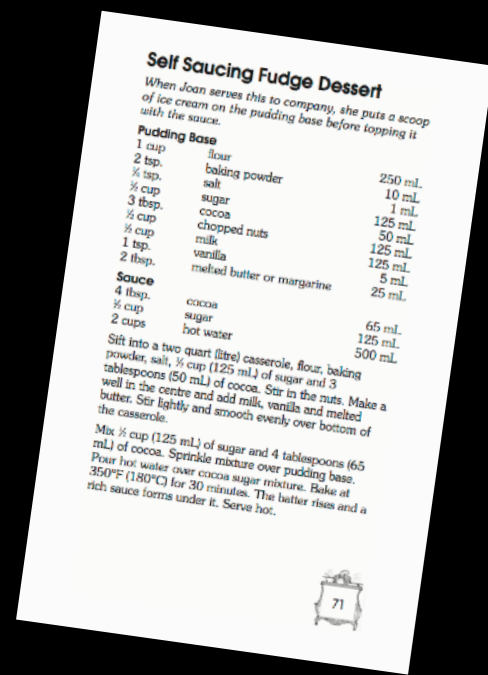
Keys Present in Recipes (by Relationship)

Key	Parent	Child	Override	Stub
Description	✓	✓		✓
Identifier	✓	✓	✓	✓
Input	optional	optional	optional	often empty
Minimum Version	✓	✓		✓
ParentRecipe	sometimes	✓	✓	
Process	✓	✓		empty array
ParentRecipeTrustInfo			✓ (optional)	

Firefox recipes

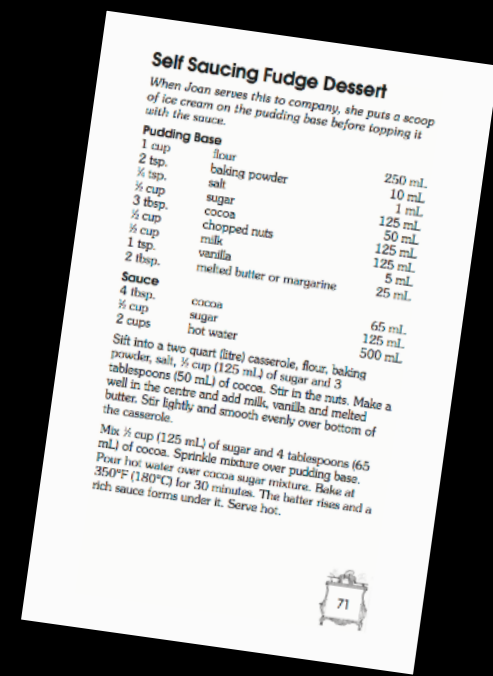


*<https://github.com/autopkg/recipes/tree/master/Mozilla>
or `~/Library/AutoPkg/RecipeRepos/com.github.autopkg.recipes/Mozilla`*



Variables

- Input variables (e.g., NAME)
- Processor variables (e.g., url)
- Substitution syntax: %NAME%
- Multiple declarations of Input variables



Common Download Recipe Workflow

1. Determine the URI of the item we want to download
2. Download a copy of the item
3. EndOfCheckPhase
4. Check the code signature of the downloaded item
 - Sometimes: Collect version information

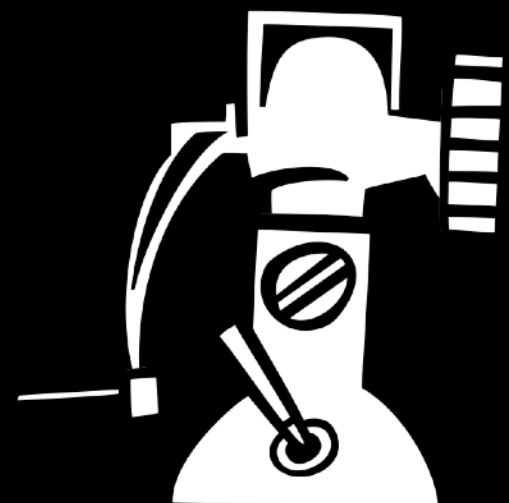
```
% autopkg run -vv Firefox.download
```



Firefox_dmg.pkg



Our first recipe!



Creating a New Recipe

- ~~Steal~~ Adapt an existing recipe
- Recipe Robot
- Write it from scratch





Creating a New Recipe

- ~~Steal~~ Adapt an existing recipe
- Recipe Robot
- Write it from scratch



```
autopkg new-recipe recipeName.function.recipe
```

Dear Kate,

Here's to the crazy ones. The mis-
rebels. The troublemakers. The ones
the square holes. The ones who see
things differently.

They're not fond of rules. And they have no
respect for the status quo.

You can praise them, disagree with them, quote
them, disbelieve them, glorify or vilify them.
About the only thing you can't do is ignore
them. Because they change things.

They invent. They imagine. They heal. They
explore. They create. They inspire. They push
the human race forward.

Maybe they have to be crazy. How else can you
stare at an empty canvas and see a work of art?
Or sit in silence and hear a song that's never
been written? Or gaze at a red planet and see a
laboratory on wheels? We make tools for these
kinds of people.

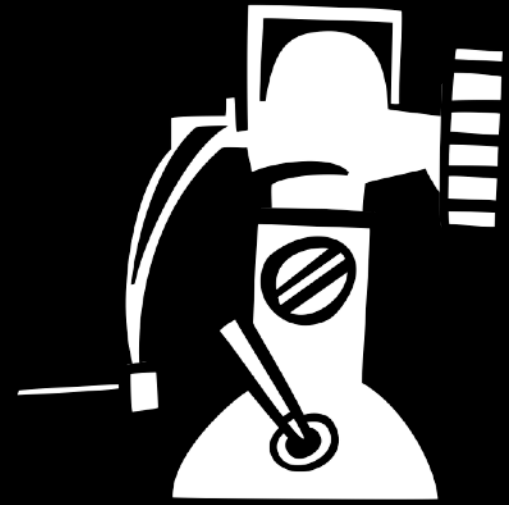
While some see them as the crazy ones, we see
genius. Because the people who are crazy enough
to think they can change the world, are the
ones who do.

Take care,
John Appleseed

Firefox.pkg

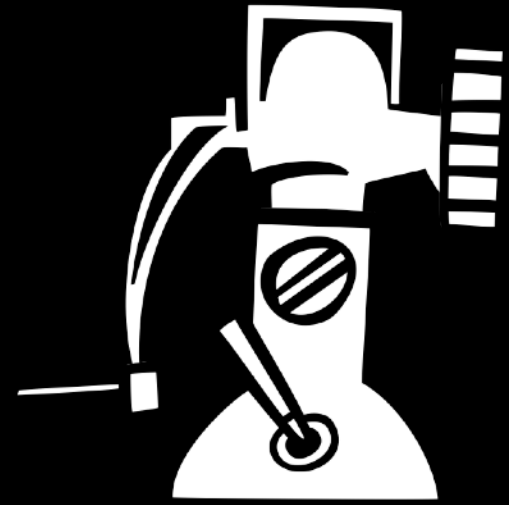
Firefox_dmg.pkg





Processors

- Python class
- Input and/or Output
- Variables often have defaults
- Core, Custom, Shared



Core Processors

- **Locally:**
 - ▶ */Library/AutoPkg/autopkglib*
- **Project:**
 - ▶ *github.com/autopkg/autopkg/tree/master/Code/autopkglib*
- **List:** `autopkg list-processors`
- **Highly trusted**
- **Documented** (Wiki or `autopkg processor-info`)

Description

Calls autopkgserver to create a package from an application.

Input Variables

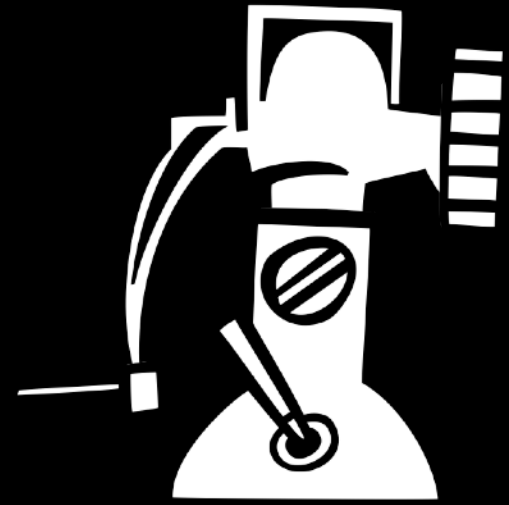
- **pkg_path:**
 - **required:** False
 - **description:** The pathname for the pkg to be created. If not set, defaults to %RECIPE_CACHE_DIR%/app_name%-version%.pkg
- **app_path:**
 - **required:** False
 - **description:** Path to an application to be packaged. Can be on a disk image and globbed. If not set, defaults to %pathname%/*.app. Typically %pathname% points to a disk image downloaded in a prior recipe step.
- **version:**
 - **required:** False
 - **description:** Version of the app. If not set, will be extracted from the CFBundleShortVersionString in the app's Info.plist.
- **force_pkg_build:**
 - **required:** False
 - **description:** When set, this forces building a new package even if a package already exists in the output directory with the same identifier and version number. Defaults to False
- **bundleid:**
 - **required:** False
 - **description:** Bundle identifier of the app. If not set, will be extracted from the CFBundleIdentifier in the app's Info.plist.

Output Variables

- **app_pkg_creator_summary_result:**
 - **description:** Description of interesting results.
- **version:**
 - **description:** Version of the app.
- **new_package_request:**
 - **description:** True if a new package was actually requested to be built. False if a package with the same filename, identifier and version already exists and thus no package was built (see 'force_pkg_build' input variable.)

Table of Contents

- [Introduction](#)
- [Getting Started](#)
- [FAQ](#)
- [More Resources](#)
- **AutoPkg Reference**
 - [Preferences](#)
 - **Recipes**
 - [Recipe Format](#)
 - [Input Variables](#)
 - [Important Variable Names](#)
 - [Recipe Search Order](#)
 - [Recipe Naming Conventions](#)
 - [Recipe Overrides](#)
 - [Recipe-writing Guidelines](#)
 - [Using CodeSignatureVerification](#)
 - [Defining Custom Variables](#)
 - [Finding Recipes](#)
 - [Sharing Recipes](#)
 - [Parent Recipe Trust Information](#)
 - [Running Multiple Recipes](#)
 - **Processors**
 - [Processor Locations](#)
 - [Pre- and postprocessors](#)
 - [Processor Summary Reporting](#)
 - [Downloading from the Internet in Custom Processors with URLGetter](#)
 - [Munki Repo Plugin Support](#)
 - [Noteworthy Shared Processors](#)
 - [Developing Custom Processors](#)
 - **Processor Reference**
 - [AppDmgVersioner](#)
 - [AppPkgCreator](#)
 - [BrewCaskInfoProvider](#)
 - [CodeSignatureVerifier](#)
 - [Copier](#)
 - [CURLDownloader](#)



Custom Processors

- When the core processors just won't do
- To share, add a (stub) recipe
 - ▶ *To call the processor from any recipe:*
`RecipeIdentifier/ProcessorName`
- "Less trusted"

Dear Kate,

Here's to the crazy ones. The misfits. The rebels. The troublemakers. The round pegs in the square holes. The ones who see things differently.

They're not fond of rules. And they have no respect for the status quo.

You can praise them, disagree with them, quote them, disbelieve them, glorify or vilify them. About the only thing you can't do is ignore them. Because they change things.

They invent. They imagine. They heal. They explore. They create. They inspire. They push the human race forward.

Maybe they have to be crazy. How else can you stare at an empty canvas and see a work of art? Or sit in silence and hear a song that's never been written? Or gaze at a red planet and see a laboratory on wheels? We make tools for these kinds of people.

While some see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Take care,
John Appleseed

Firefox_dmg.pkg



Syntax



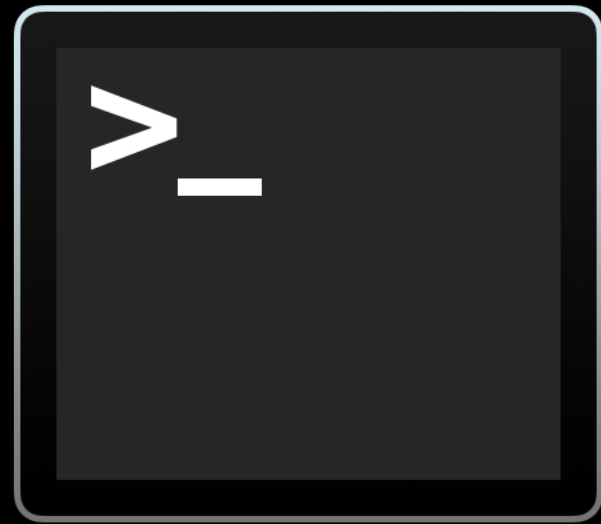
```
plutil -lint /path/to/recipe
```



Markup > Check > Document Syntax

^⌘Y

Standards



```
plutil -convert xml1 /path/to/recipe
```



Text > Apply Text Filter > *Filter Name*

```
plutil -convert xml1 - -o -
```

Dear Kate,

Here's to the crazy ones. The misfits. The rebels. The troublemakers. The round pegs in the square holes. The ones who see things differently.

They're not fond of rules. And they have no respect for the status quo.

You can praise them, disagree with them, quote them, disbelieve them, glorify or vilify them. About the only thing you can't do is ignore them. Because they change things.

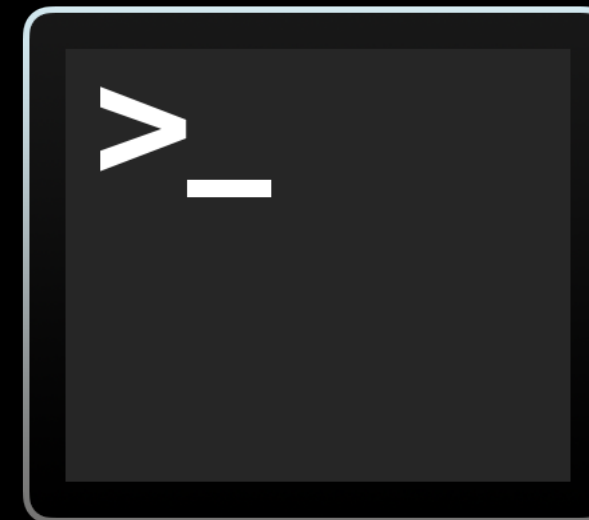
They invent. They imagine. They heal. They explore. They create. They inspire. They push the human race forward.

Maybe they have to be crazy. How else can you stare at an empty canvas and see a work of art? Or sit in silence and hear a song that's never been written? Or gaze at a red planet and see a laboratory on wheels? We make tools for these kinds of people.

While some see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Take care,
John Appleseed

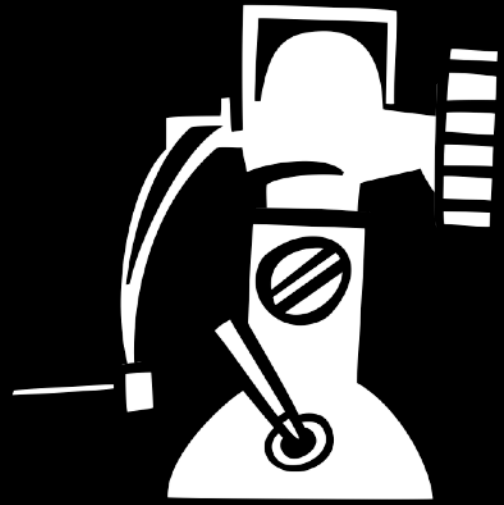
Firefox_dmg.pkg





Testing 1, 2, 3

1. Test the parent
2. Test my new child recipe
3. Test it again



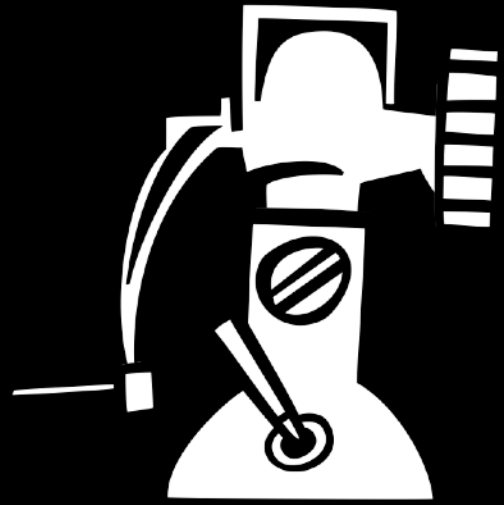
Strict Trust Verification

- Set Preference


- ▶ `defaults write com.github.autopkg
FAIL_RECIPES_WITHOUT_TRUST_INFO -bool YES`

- Wiki

- ▶ *<https://github.com/autopkg/autopkg/wiki/AutoPkg-and-recipe-parent-trust-info>*



Ignore Trust Info

- Persistent
 - ▶ `defaults write com.github.autopkg FAIL_RECIPES_WITHOUT_TRUST_INFO -bool NO`
 - ▶ `defaults delete com.github.autopkg FAIL_RECIPES_WITHOUT_TRUST_INFO`
- Per Run 
 - ▶ `autopkg run -vv --ignore-parent-trust-verification-errors Product.pkg`

```
% alias aprt='autopkg run -vv --ignore-parent-trust-  
verification-errors'
```

*Aliases stored in
~/.bash_profile or ~/.bashrc or ~/.zshrc*

```
% autopkg run -vv --ignore-parent-trust-verification-errors  
Firefox.download  
[results shown earlier]  
% autopkg run -vv --ignore-parent-trust-verification-errors  
Firefox_dmg.pkg  
[...]  
% autopkg run -vv --ignore-parent-trust-verification-errors  
Firefox_dmg.pkg  
[...]
```



FirefoxSignedPkg



*<https://github.com/autopkg/recipes/tree/master/Mozilla>
or `~/Library/AutoPkg/RecipeRepos/com.github.autopkg.recipes/Mozilla`*

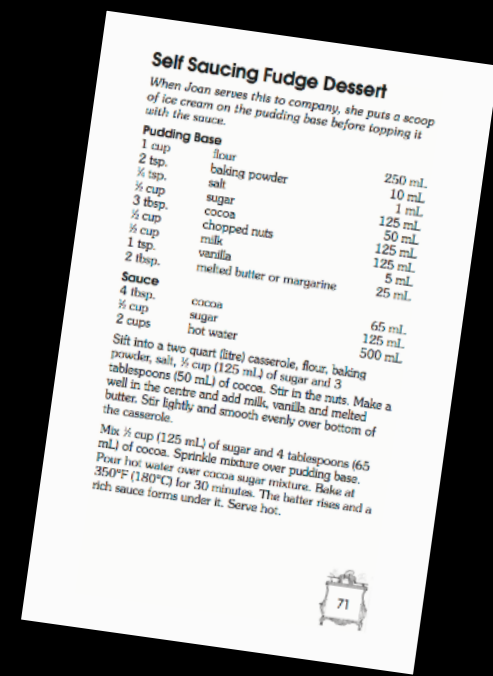
Workshop Outline

We're almost home!

- ✓ Prerequisites
- ✓ Recipe Reading & Literacy
- ✓ Creating a new recipe
- ✓ Testing your recipe

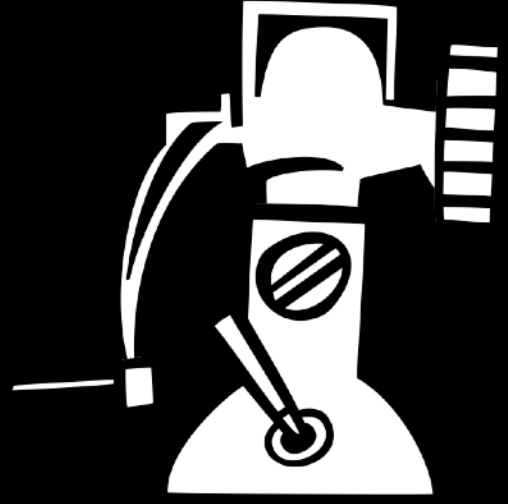
Writing a download recipe LIVE!





Common Download Recipe Workflow

1. Determine the URI of the item we want to download
2. Download a copy of the item
3. EndOfCheckPhase
4. Check the code signature of the downloaded item
 - Sometimes: Collect version information

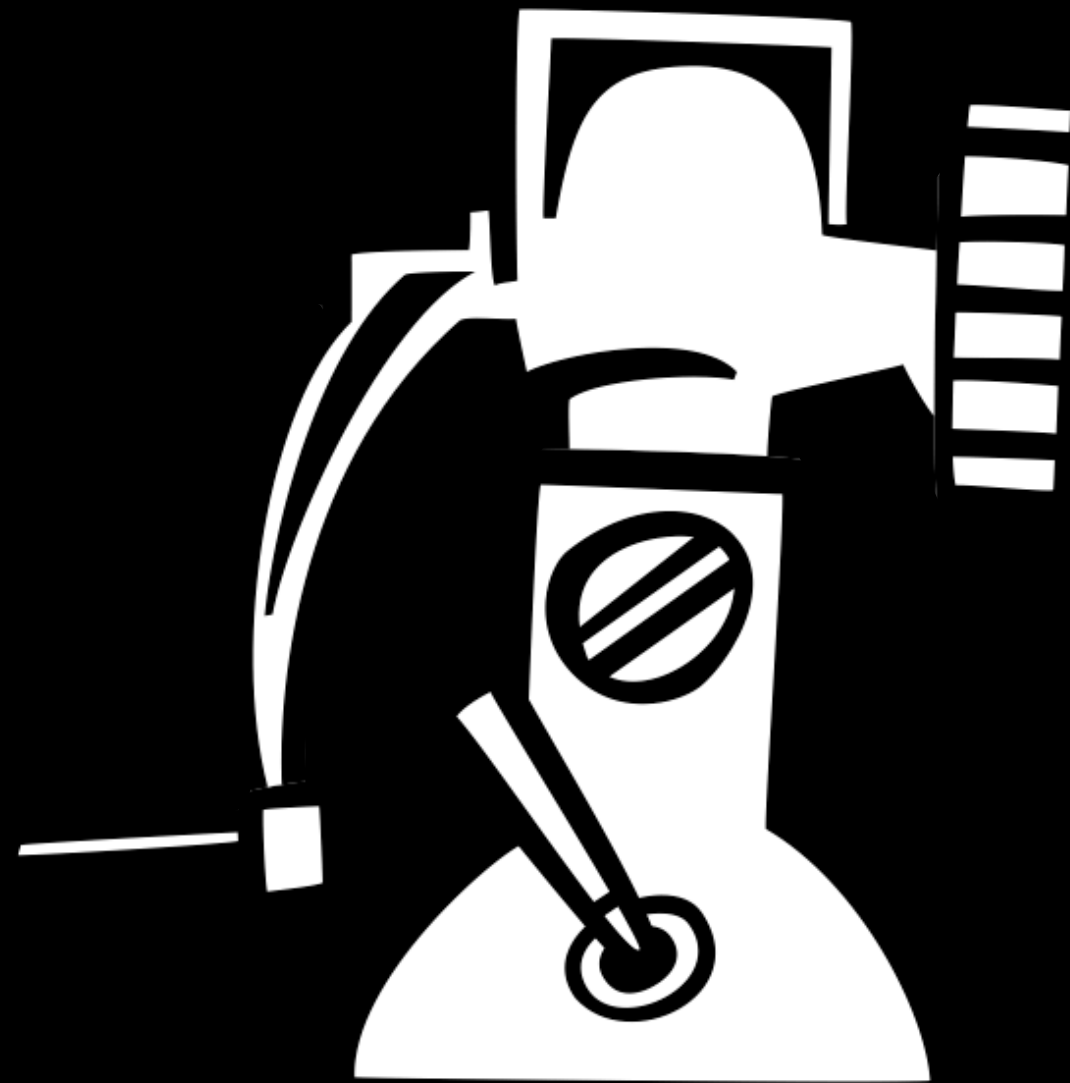


Retrieve Code Signature

- Application Bundle (.app)
 - ▶ `codesign --display -r- --deep /path/to/the.app`
 - ▶ *If .zip, unarchive first (Unarchiver processor in recipe)*
- Package Installer (.pkg)
 - ▶ `pkgutil --check-signature /path/to/the.pkg`

<https://github.com/autopkg/autopkg/wiki/Using-CodeSignatureVerification/>

Writing a download recipe **LIVE!**





Wrapping Things Up

<https://maclabs.jazzace.ca/2020/11/26/autopkg-workshop.html>

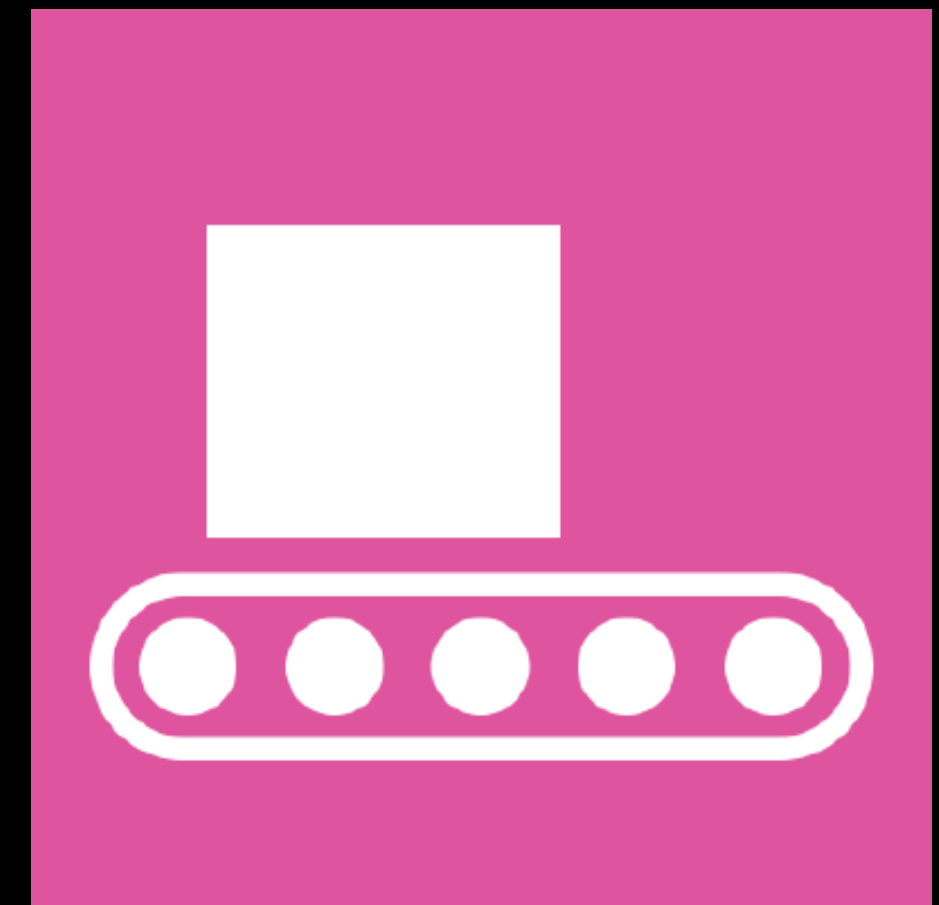
Anthony Reimer

 maclabs.jazzace.ca

 [AnthonyReimer](https://twitter.com/AnthonyReimer)

 [jazzace](https://jazzace.ca)

 [jazzace](https://github.com/jazzace)



**UNIVERSITY OF
CALGARY**