

# Splunk App Lifecycle Management – with More Peace, Love and Rock-n-Roll!

Grigori Melnik and Cecelia Campbell

Splunk

.conf2016

splunk>

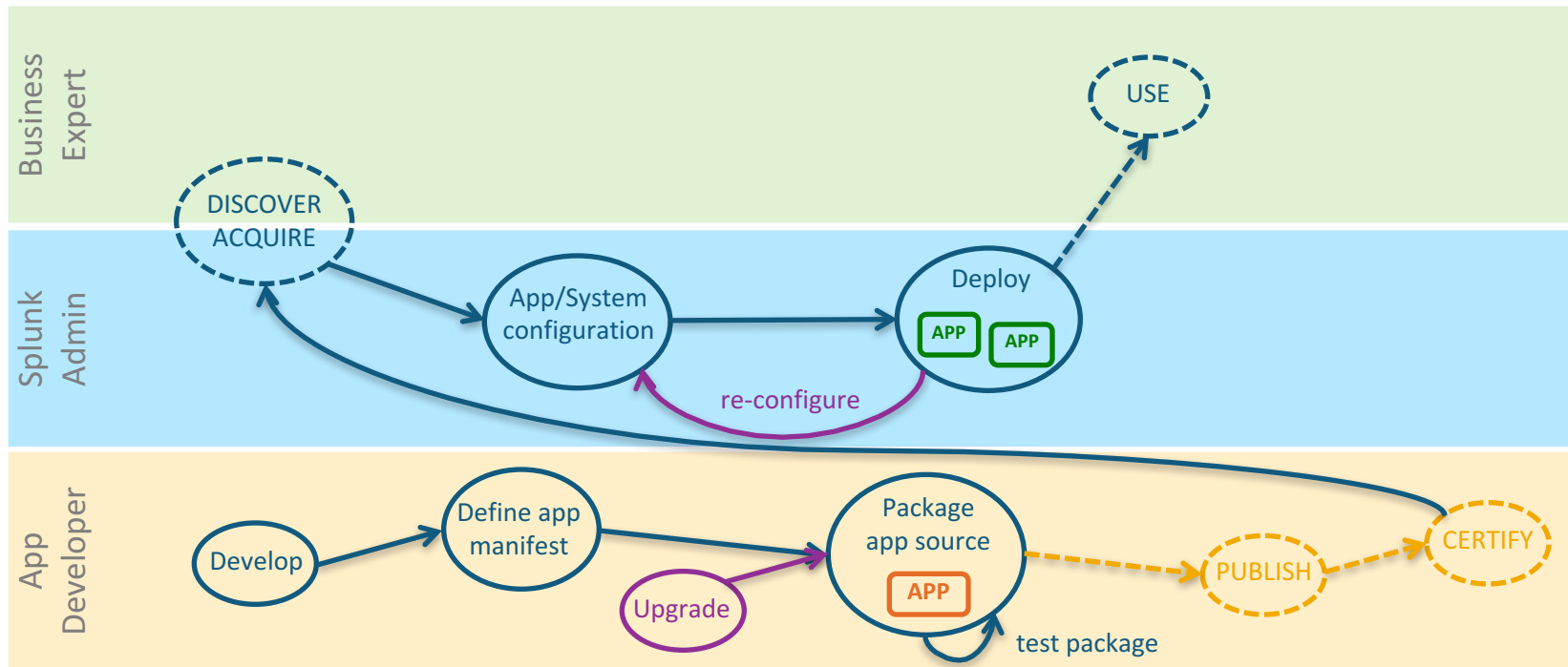
# Disclaimer

During the course of this presentation, we may make forward looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC. The forward-looking statements made in the this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not, be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

# App Manageability

- 1 As a **developer**, I want to lower my cost of Splunk app development – let me focus on building apps without concerning myself about the deployment topologies and the nitty-gritty of the deployment process.
- 2 As an **admin**, I want to easily & reliably install and manage any kind of content (apps, addons, modules, content packs) across my entire Splunk deployment.

# Targeted User Flow



# The poetry of app manageability

New app **packaging & deployment**  
**tools** and **guidance**

for **developers** and **admins** that

**simplify**

app **deployment** and **troubleshooting** to

**distributed** environments,

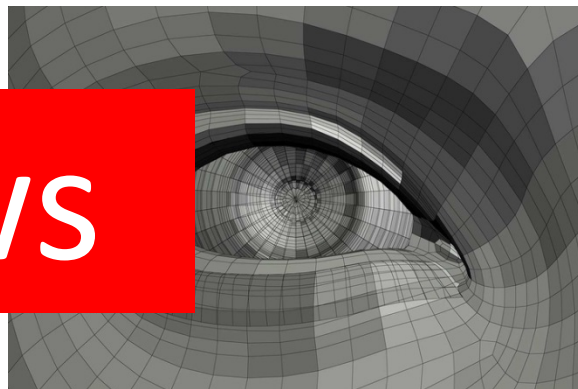
while preserving app **backward compatibility**

with existing tools & practices.

# The mechanics of app manageability

- Focus on **disambiguating** config and *partitioning* (packaging) relevant pieces of config+code into **deployment packages**
  - along **physical** workloads and **logical** groups of forwarders

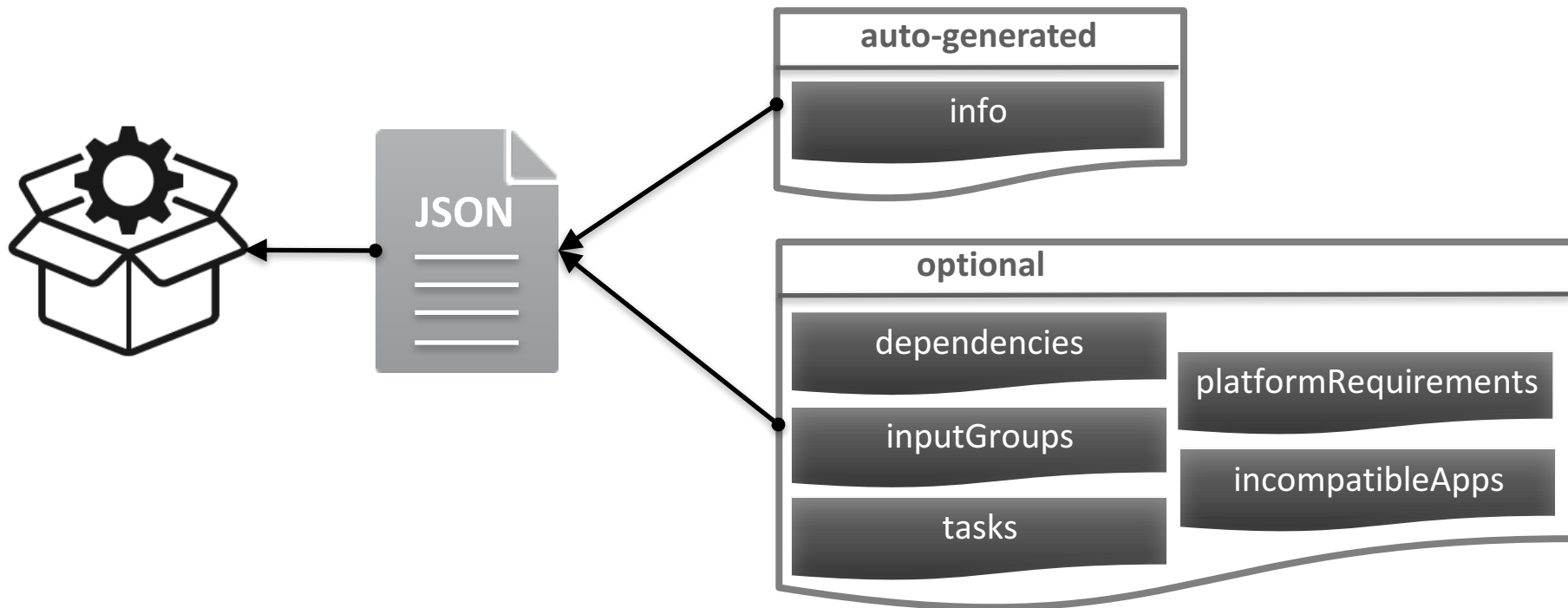
The system knows



# The Packaging Toolkit

- All-in-one tool for both developers and admins
1. Devs use it to define and package an app
  2. Admins use it to partition and prepare for deployment
  3. Splunk platform (future) mechanism will deploy the partitioned app

# Packaging with an App Manifest

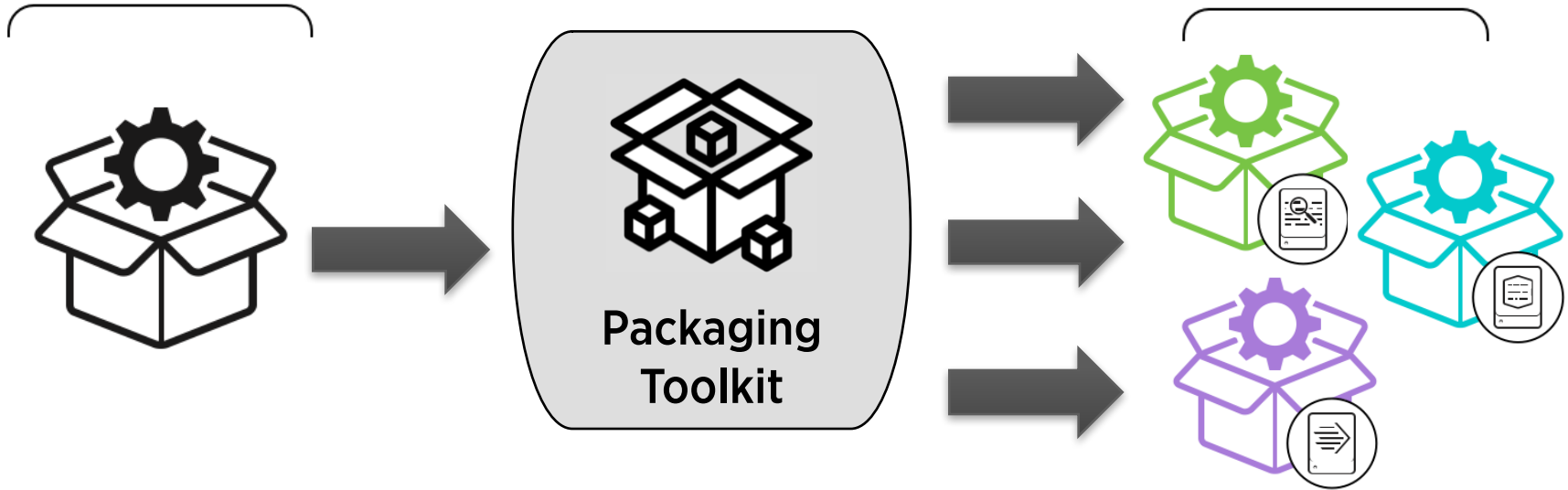




# Partitioning a Packaged App

Source package

Deployment Packages



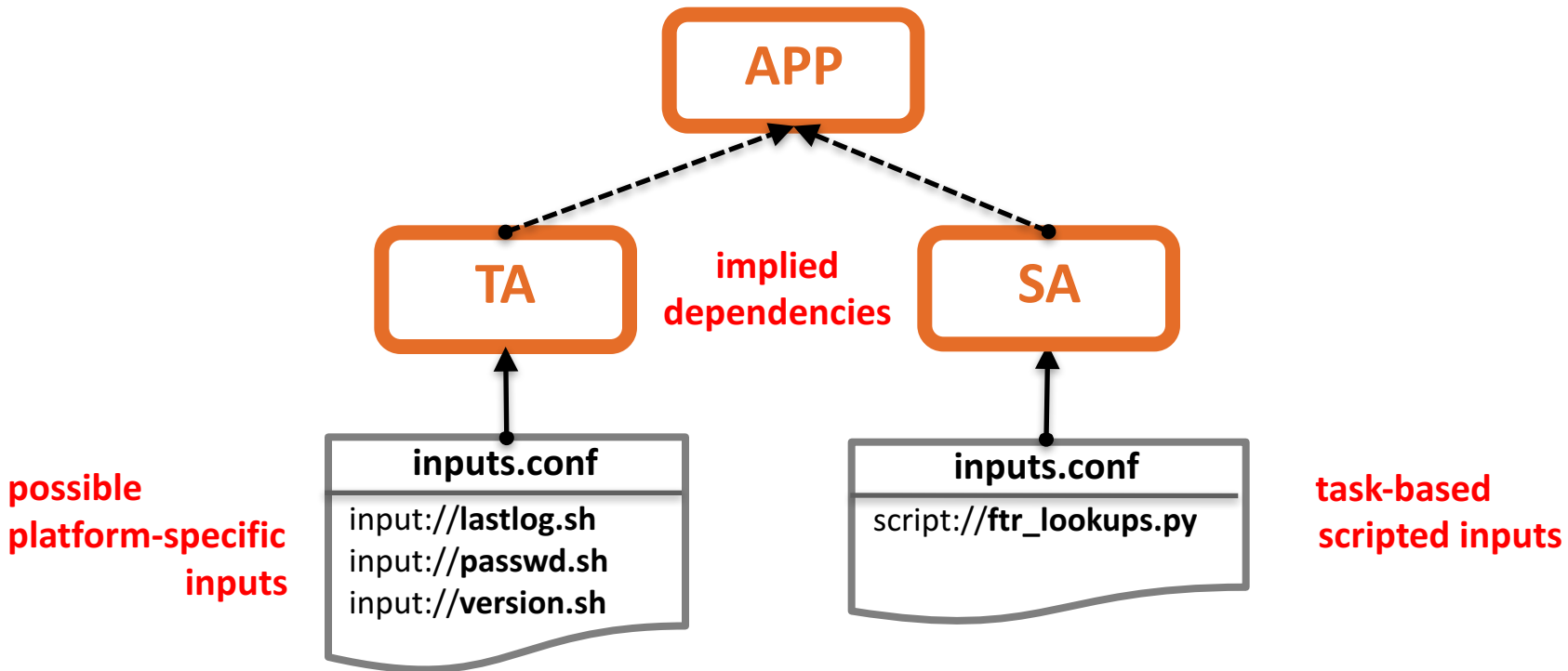
# Packaging Toolkit Commands Cheat Sheet

- **generate-manifest** – generate a manifest for an app based on its conf
- **package** – create a source package with manifest
- **partition** – partition an app into a set of targeted deployment packages
- **describe** – describe an app configuration & dependencies
- **validate** – validate an app content (incl. app manifest, packaged dependencies, well-formedness)

# Dev Flow Example



# Sample App – Splunk App For \*Nix



# Generate a Manifest

```
$ slim generate-manifest SA_nix -o SA_nix/app.manifest
slim generate-manifest: Parsing app configuration at "SA_nix"...
slim generate-manifest: Generating app manifest to "SA_nix/app.manifest"...
slim generate-manifest: [NOTE] App manifest saved to "SA_nix/app.manifest"

$ slim generate-manifest TA_nix -o TA_nix/app.manifest
...

$ slim generate-manifest splunk_app_for_nix -o splunk_app_for_nix/app.manifest
...
```

# App Manifest - info

```
"info": {
  "title": "...",
  "id": { ... },
  "author": { ... },
  "releaseDate": "...",
  "description": "...",
  "license": { ... },
  "releaseNotes": { ... }
}
```

# App Manifest - examples

```
# Define dependencies and versions to enforce
# "dependencies": {
#   "<app-id>": {
#     "version": "*",
#     "package": "<source-package-location>"
#   }
# }
#
# Define inputs that are management tasks
# "tasks": []
```

```
# Define custom and dependency input groups
# "inputGroups": {
#   "<group-name>": {
#     "requires": {
#       "<app-id>": ["<input-group-name>"]
#     },
#     "inputs": ["<defined-inputs>"]
#   }
# }
```

# App Manifest – SA\_nix

```
"tasks": [  
    "script://./bin/scripted_inputs/ftr_lookups.py",  
]
```



# App Manifest – TA\_nix

```
"inputGroups": {  
  "User Monitoring": {  
    "description": "Monitor current user sessions and login history",  
    "inputs": ["script://./bin/who.sh", "script://./bin/lastlog.sh"]  
  },  
  "OSX Inputs": {  
    "description": "ES scripted inputs supported on only OSX platforms",  
    "inputs": ["script://./bin/sshdChecker.sh"]  
  },  
  "Linux Inputs": {  
    "description": "ES scripted inputs supported on Linux platforms",  
    "inputs": ["script://./bin/selinuxChecker.sh"]  
  }  
}
```

# App Manifest – splunk\_app\_for\_nix

```
"dependencies": {  
  "SA_nix": {  
    "version": "~5.2.0",  
    "package": "SA_nix-5.2.0.tar.gz"  
  },  
  "TA_nix": {  
    "version": "^5.2.0",  
    "package": "TA_nix-5.2.3.tar.gz"  
  }  
}
```

# Create a Source Package

```
$ slim package SA_nix
```

```
slim package: Packaging app at "SA_nix"...
```

```
slim package: [NOTE] Source package exported to "SA_nix-5.2.0.tar.gz"
```

```
$ slim package TA_nix
```

```
slim package: Packaging app at "TA_nix"...
```

```
slim package: [NOTE] Source package exported to "TA_nix-5.2.3.tar.gz"
```

```
$ slim package splunk_app_for_nix
```

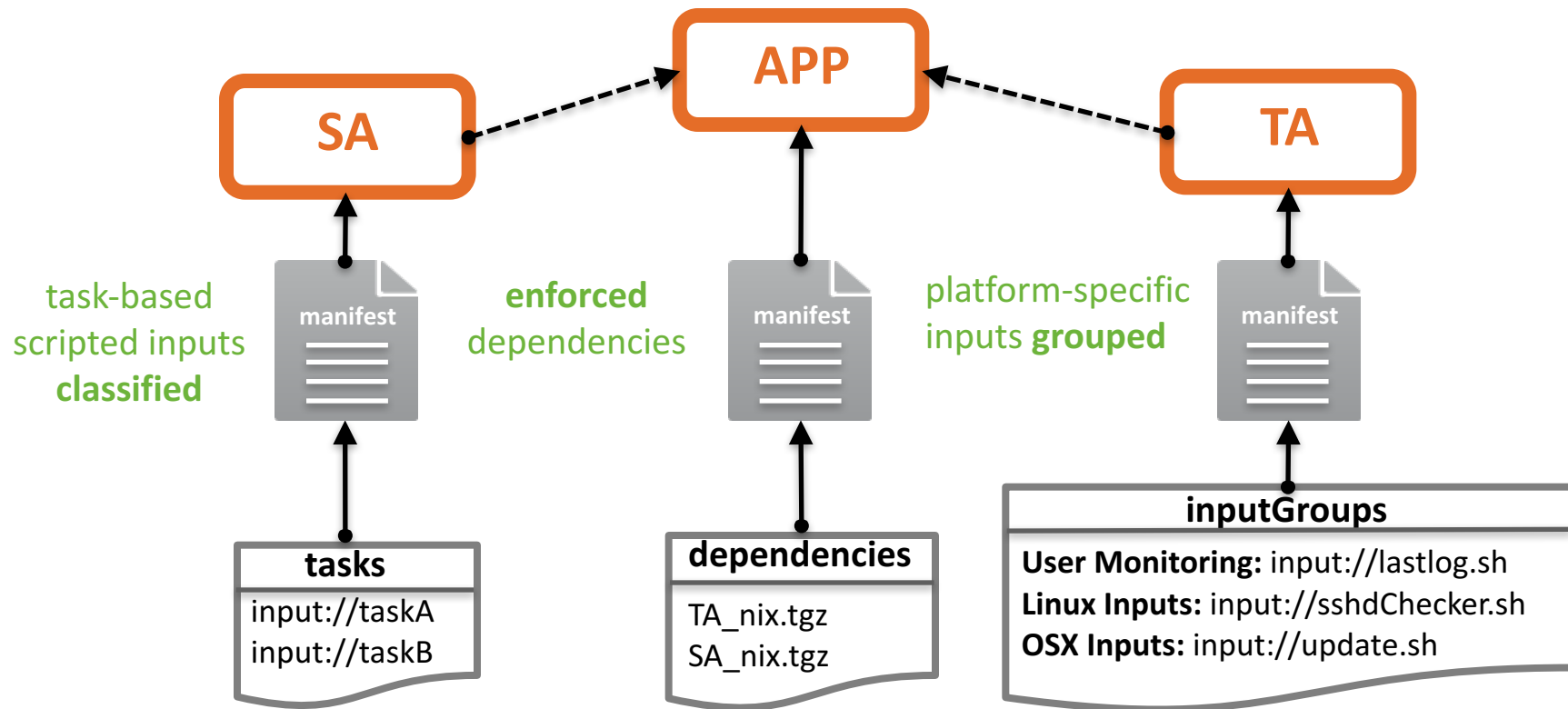
```
slim package: Packaging app at "splunk_app_for_nix"...
```

```
slim package: [NOTE] Source package exported to "splunk_app_for_nix-5.2.0.tar.gz"
```

# Describe an App Package

```
$ slim describe splunk_app_for_nix-5.2.0.tar.gz
slim describe: Describing "splunk_app_for_nix-5.2.0.tar.gz"...
[info]
|-- The Splunk App for Unix offers new ways to alert, report, and investigate data.
|   |-- by Splunk, Inc.
|   |-- defined as splunk_app_for_nix version 5.2.0
[input-groups]
|-- User Monitoring defines no inputs and requires [TA_nix]
|-- Linux Group defines no inputs and requires [TA_nix]
|-- SunOS Group defines no inputs and requires [TA_nix]
|-- OSX Group defines no inputs and requires [TA_nix]
[dependency-graph]
|-- splunk_app_for_nix@5.2.0
|   |-- SA_nix@5.2.0 (accepting ~5.2.0)
|   |-- TA_nix@5.2.3 (accepting ^5.2.0)
```

# Sample App – Updated




# Benefits for Devs

	Previous Method	Packaging Toolkit
App Info	<ul style="list-style-type: none"><li>• Scattered across conf</li></ul>	<ul style="list-style-type: none"><li>• Centralized location</li><li>• Automatically generated</li></ul>
Dependency Management	<ul style="list-style-type: none"><li>• Release Notes required</li><li>• Guessing version compatibilities</li></ul>	<ul style="list-style-type: none"><li>• Defined and Enforced</li><li>• SemVer compatible</li></ul>
Input Groups	<ul style="list-style-type: none"><li>• All content, everywhere</li></ul>	<ul style="list-style-type: none"><li>• Logically grouped</li></ul>
Management Tasks	<ul style="list-style-type: none"><li>• Undefined</li></ul>	<ul style="list-style-type: none"><li>• Treated differently</li></ul>

# Call to Action for Devs!

- Start onboarding your apps with the Packaging Toolkit
- Generate a manifest and customize your requirements
- Give us feedback : [AppMgmt-feedback@splunk.com](mailto:AppMgmt-feedback@splunk.com)

- # Call to Action for Devs!
- Start onboarding your apps with the Packaging Toolkit
  - Generate a manifest and customize your requirements
  - Give us feedback : [AppMgmt-feedback@splunk.com](mailto:AppMgmt-feedback@splunk.com)



But  
wait,  
in the  
future  
...



# Benefits for Admins

- Dependencies are explicitly declared by the Devs
  - Admins can view and reconcile app dependencies to avoid conflicts ***automatically***
- Inputs are logically grouped by the Devs
  - Admins can target specific logic to the appropriate workloads ***automatically***

Even without an app manifest, the Packaging Toolkit will be able to partition based on a default set of rules!

# Admin Flow PREVIEW



# Apps

A Splunk app is an application that runs on Splunk Enterprise or Splunk Cloud and typically addresses several use cases. It can include Splunk Enterprise knowledge objects such as reports, lookups, and inputs.

0 Apps

filter

20 per page ▾

Install App

Create App

Name	Actions	Folder Name ^	Version	Context	After Installation	Instances Up to Date	Pending Change
------	---------	---------------	---------	---------	--------------------	----------------------	----------------

! No apps found.

Install App

Select App

Set Properties

Review

Done

< Next >

Select App File

Choose a file by either ~~browsing your~~ computer or dropping the file into the target box below.

Selected file: **splunk\_app\_for\_nix-5.2.0.tar.gz**

Select App File



Drop your app file here

Install App

Select App Set Properties Review Done

< Review >

Input Groups

Map the following logical input groups defined by the app to your target deployment topology.

User Monitoring

No Forwarders

All forwarders

Selected server classes

× Europe

× North America

SunOS Group

No Forwarders

All forwarders

Selected server classes

Linux Group

No Forwarders

All forwarders

Selected server classes

OSX Group

No Forwarders

All forwarders

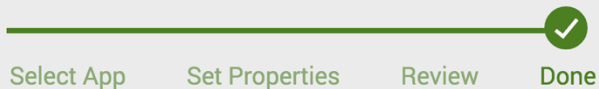
Selected server classes

After installation

Select an action that will be performed on all selected forwarders once app installation is complete.

Do nothing Restart Splunk Reload App

## Install App



Next &gt;



Application was installed successfully.

Configure your apps by going to [Apps](#)

## Apps

[Install App](#)[Create App](#)

A Splunk app is an application that runs on Splunk Enterprise or Splunk Cloud and typically addresses several use cases. It can include Splunk Enterprise knowledge objects such as reports, lookups, and inputs.

3 Apps

20 per page ▾

Name	Actions	Folder Name ^	Version	Context	After Installation	Instances Up to Date	Pending Change
<a href="#">SA_nix</a>	<a href="#">Edit ▾</a> <a href="#">Manage Inputs</a> <a href="#">Export</a>	SA_nix	5.2.0	All Forwarders	Do nothing	0/0 ( <a href="#">details</a> )	<a href="#">See Details</a>
<a href="#">Splunk Add-on for *Nix</a>	<a href="#">Edit ▾</a> <a href="#">Manage Inputs</a> <a href="#">Export</a>	TA_nix	5.2.3	All Forwarders	Do nothing	0/0 ( <a href="#">details</a> )	<a href="#">See Details</a>
<a href="#">Splunk App for Unix</a>	<a href="#">Edit ▾</a> <a href="#">Manage Inputs</a> <a href="#">Export</a>	splunk_app_for_nix	5.2.0	All Forwarders	Do nothing	0/0 ( <a href="#">details</a> )	<a href="#">See Details</a>

Deploy Changes

Cancel Pending Changes Deploy Pending Changes

All pending changes to Server Classes, Apps, and Forwarders can be deployed by pressing Deploy Changes button. A detailed list of recently deployed changes is under the Recently Deployed Changes tab.

Pending Changes

Recently Deployed Changes

+ Expand All Changes

Last Operation

25 per page ▾

5 Changes

Entity Type: All ▾

filter by Entity Name





i	Entity Name ▾	Entity Type ▾	Operation	Edit Time ▾	Edit User ▾
>	splunk_app_for_nix	App	Install	9/26/2016, 1:17:04 AM	admin
>	SA_nix	App	Install	9/26/2016, 1:17:04 AM	admin
>	TA_nix	App	Install	9/26/2016, 1:17:04 AM	admin
>	North America	Server Class	Create	9/26/2016, 1:15:57 AM	admin
>	Europe	Server Class	Create	9/26/2016, 1:15:37 AM	admin




Create App

A Splunk app is an application that runs on Splunk Enterprise or Splunk Cloud and typically addresses several use cases. It can include Splunk Enterprise knowledge objects such as reports, lookups, and inputs.

20 per page ▾

Name	Actions			Folder Name 	Version	Context	After Installation	Instances Up to Date	Pending Change
SA_nix	Edit 	Manage Inputs	Export	SA_nix	5.2.0	All Forwarders	Do nothing	0/0 <a href="#">(details)</a>	
Splunk Add-on for *Nix	Edit 	Manage Inputs	Export	TA_nix	5.2.3	All Forwarders	Do nothing	0/0 <a href="#">(details)</a>	
Splunk App for Unix	Edit 	Manage Inputs	Export	splunk_app_for_nix	5.2.0	All Forwarders	Do nothing	0/0 <a href="#">(details)</a>	

- 
- [Edit Properties](#)  
[Edit Configuration](#)  
[Update](#)  
[Uninstall](#)

# Key Takeaways

- App manageability (installation/uninstallation/update)
  - Automatic dependency resolution (cascading)
  - Mapping of logical input groups to server classes
  - Partitioning into targeted deployment packages

## Choose your deployment mechanism

- **Now:** Chef/Ansible/Puppet/... playbook/recipe/script
- **Future:** App Management UI

# What's Next?

- Download the public beta of the Packaging Toolkit today:  
<http://dev.splunk.com/goto/packaging-toolkit>
- Come visit us at the Dev Tools & Guidance Booth!
  - Learn More and see the Demos
- Give us feedback : [AppMgmt-feedback@splunk.com](mailto:AppMgmt-feedback@splunk.com)

# THANK YOU

.conf2016