Unified Open-Sourced Splunk Configuration Management System

Vince Liggio

Splunk Team Lead, Bridgewater

Casey Pike

Consultant, Aplura

Kal Patel

Software Engineer, Bridgewater

.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.

About Bridgewater

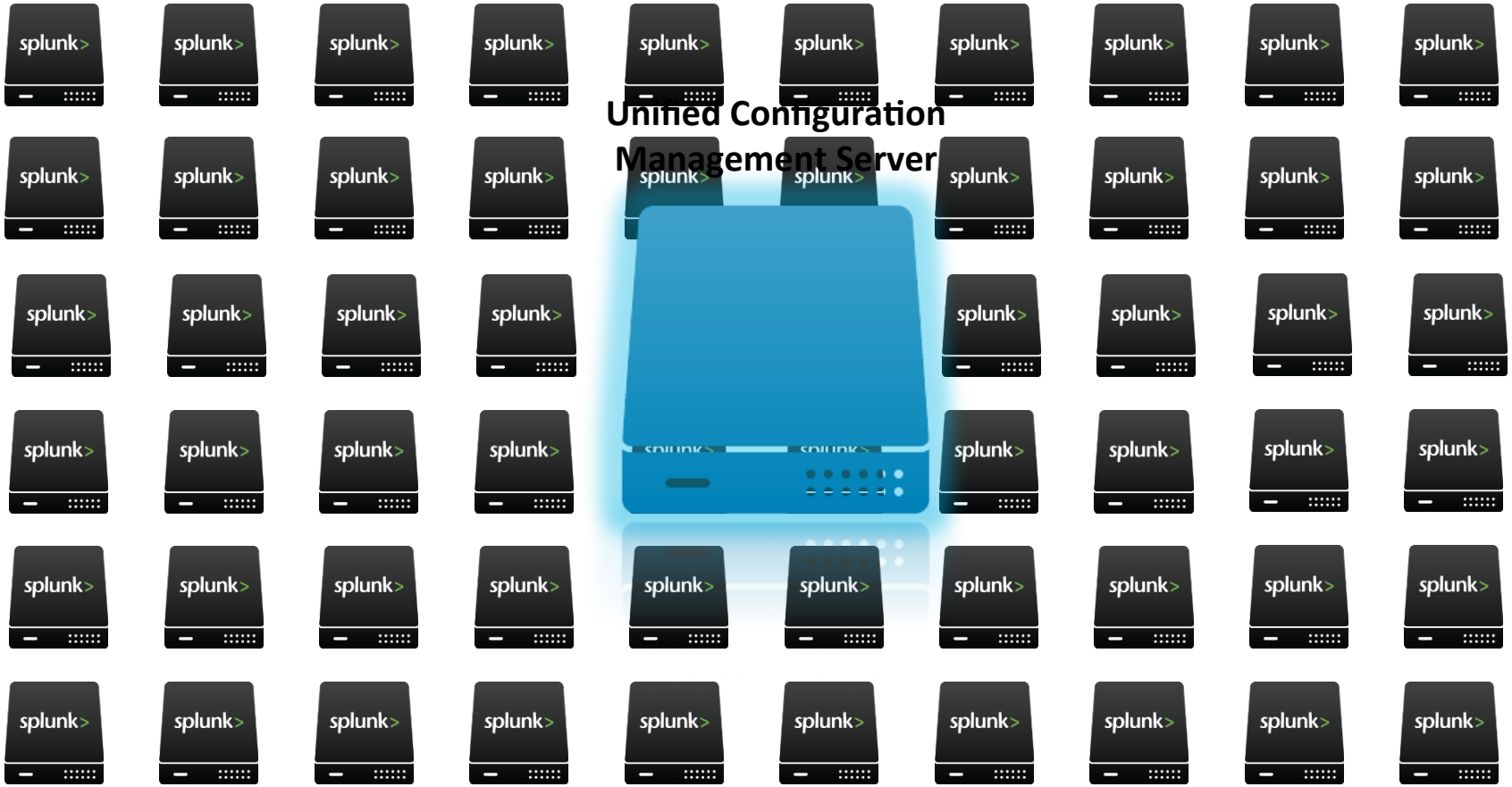
- One of the largest Hedge Funds in the world
- Manages approximately $150 billion in global investments
- Founded in 1975 out of a two-bedroom apartment
- Approximately 1,500 people work at Bridgewater
- Headquartered in Westport, Connecticut

About Us

- **Casey** – Information Systems Consultant at Aplura
  - Splunking since 2012
  - Implemented multiple multi-terabyte Splunk deployments
  - Enjoys long walks Splunking on the beach and candlelight dinners ingesting data
- **Kal** – Software engineer at Bridgewater Associates
  - Cloud engineering and automation specialist
  - Background in developing TV streaming graphics for sporting events
  - Holds multiple patents in gambling gaming technology
- **Vince** – Splunk Tech Lead at Bridgewater Associates
  - Experience in high performance computing and special effects
  - Multiple motion-picture animation movie credits (see my IMDB page)
  - Countless years delivering Unix based solutions as a VAR

Pop Quiz?

- Administered a Splunk cluster?
- Tried jury-rigging various configuration management systems to control Splunk?
- Experience pain just thinking of managing applications on deployers, deployment servers, cluster master, license master, indexers, search heads, and search head clusters?
- Know exactly what version, down to the code changes, of an application that is running on every server of your environment?



Agenda

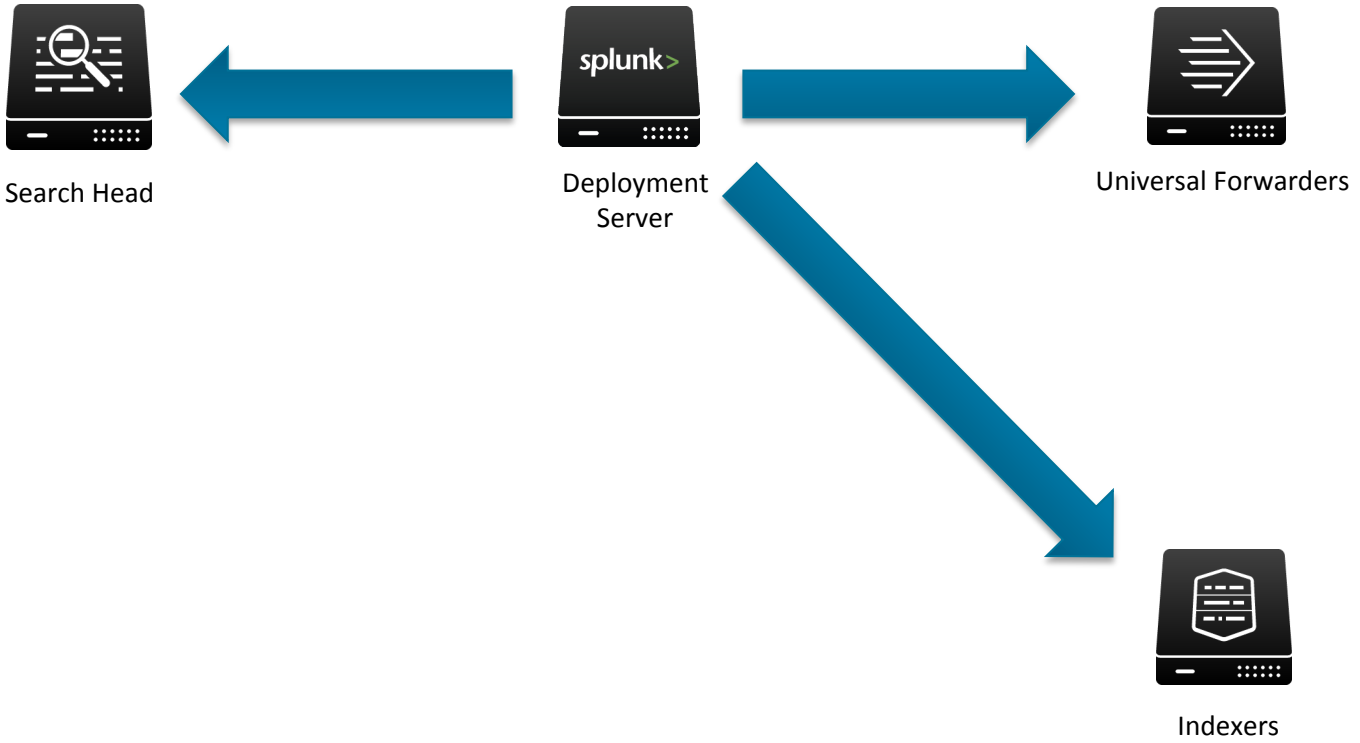
- Why Create A Unified Splunk Configuration Management System?
- What Are We Trying To Achieve?
- Overview Of What We Created
- How It Works
- Demonstration

Why Did We Do This?

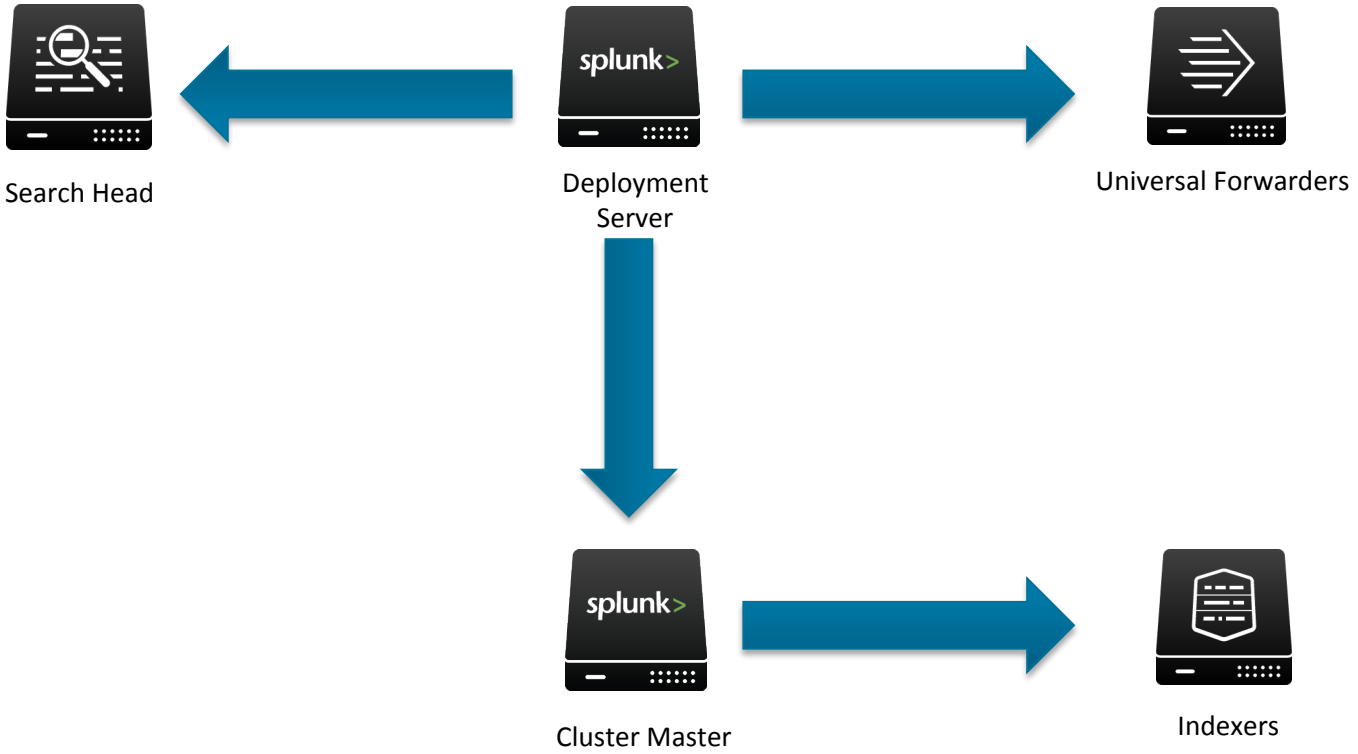
Pain

- Complex Splunk Environment
- 4 Splunk Methods of Deploying
- No Auditing
- Time Consuming
- No Quality Change Control

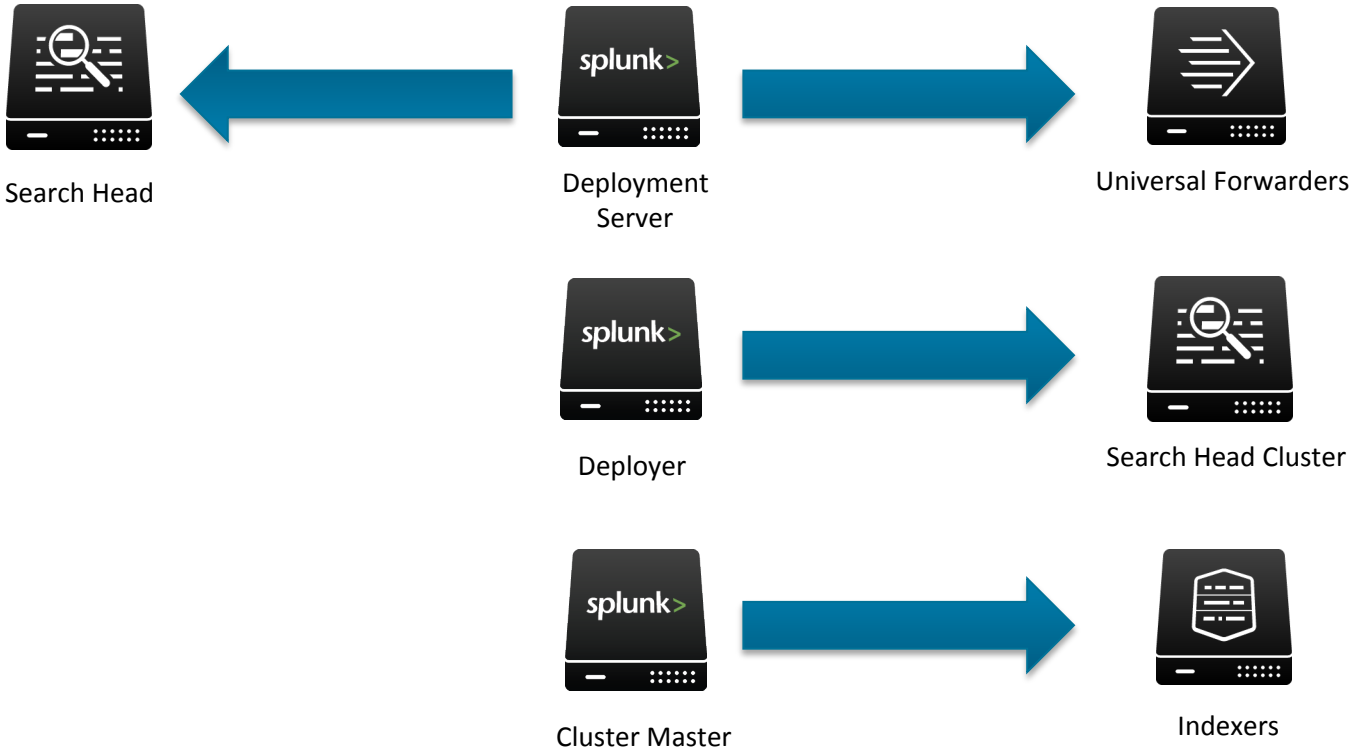
Splunk 4.x Deployment



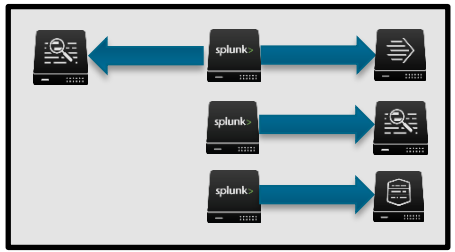
Splunk 5.x Deployment

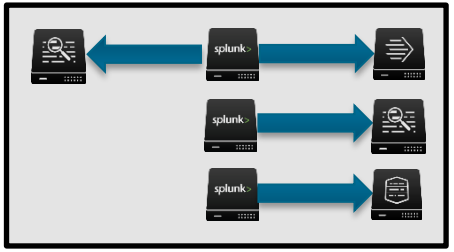


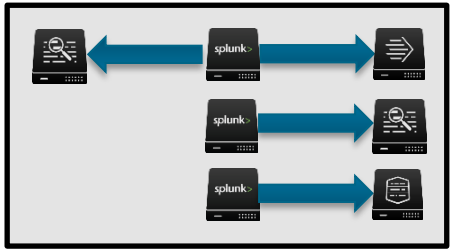
Splunk 6.x Deployment

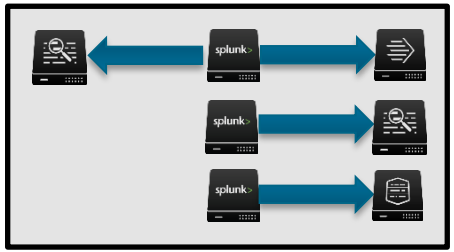


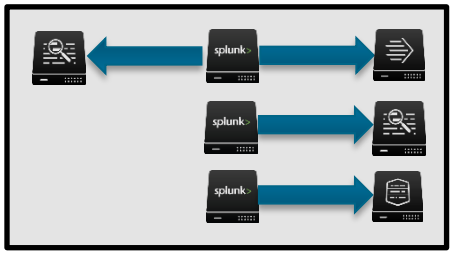
Multi-Environment Deployment











What Are We Trying To Achieve?

Goal

- Auditable tracking of all changes
- Universal deployment of applications
- Repeatable automation
- Inherent change control
- Reduced complexity
- Decreased deployment time
- Does not reinvent the wheel

What Is It?

We call it

Appetite

Appetite!

- Splunk specific python application
- Pulls applications from source control
- Creates host specific install packages
- Deploys to Splunk instances based on a manifest file
- Logs all actions

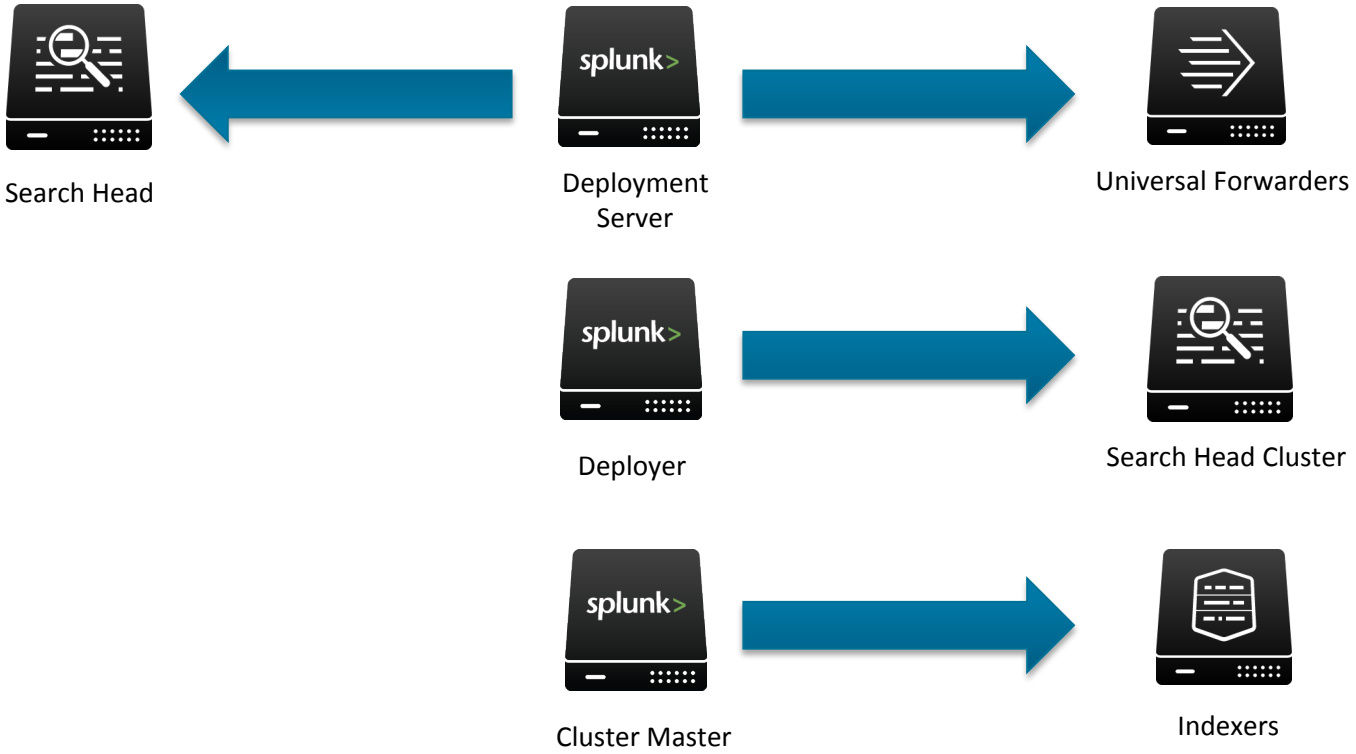
Why The Name Appetite?

- Appetite got its name from its function. The Splunk servers are fed applications from a repo… plus it has App in the name ☺.

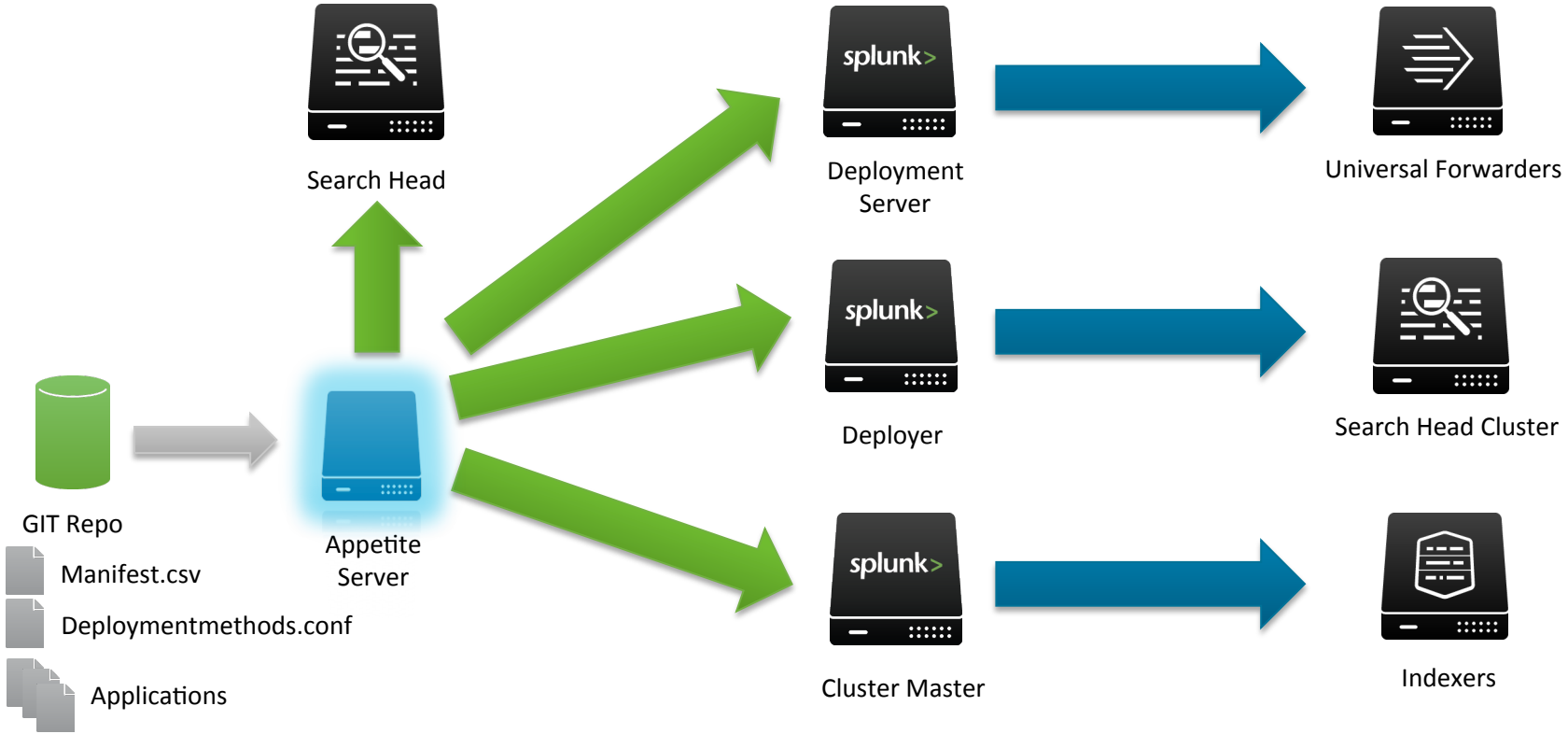
ap·pe·tite
noun
1. a natural desire to satisfy a server's need, especially for ~~food~~ applications.
"it has a healthy appetite"

How Does It Work?

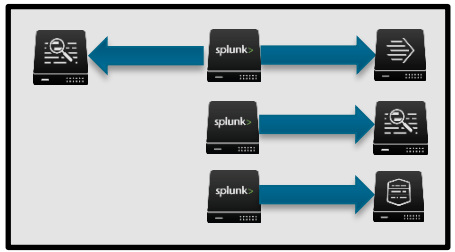
Splunk 6.x Deployment

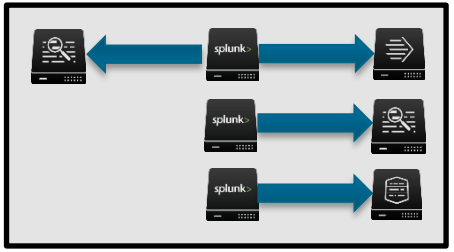


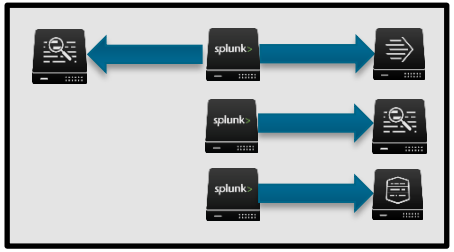
Appetite Diagram

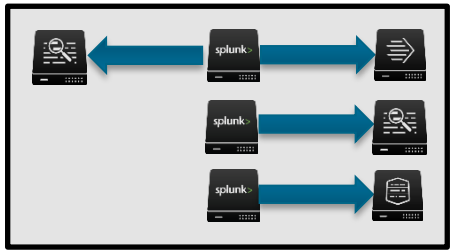


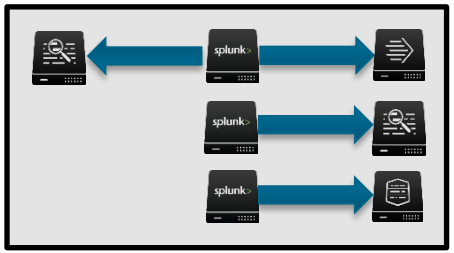
Multi-Environment Deployment



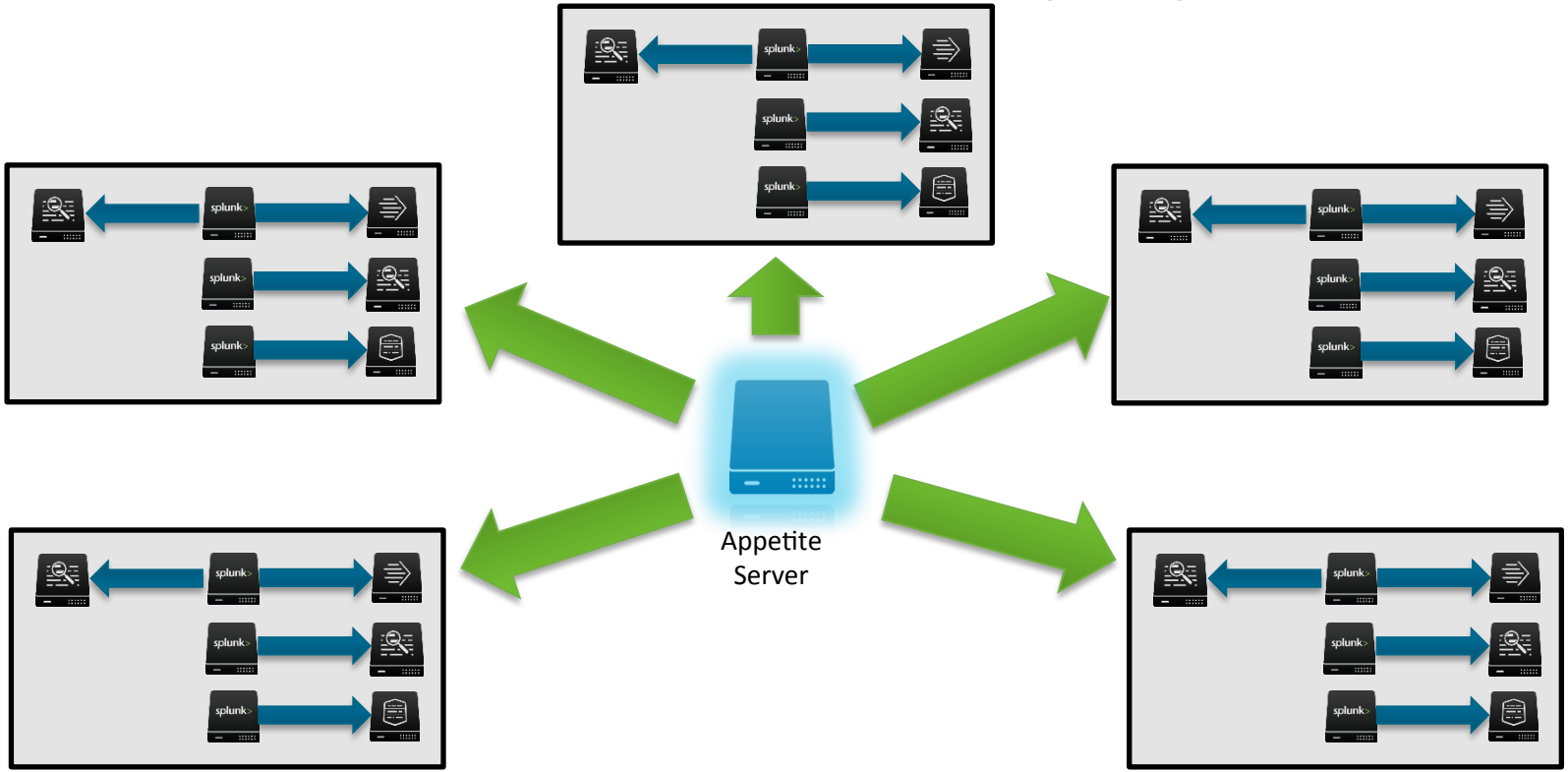








Multi-Environment Deployment

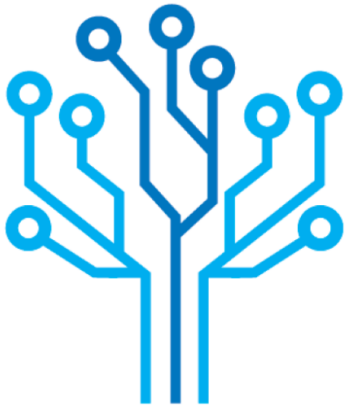


Appetite!

- Only Two Key Components
  - GIT Repository
  - Appetite Server (non-dedicated)

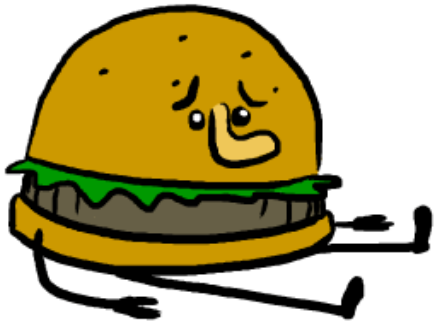
Why GIT?

- Open source distributed version control system
- All changes are individual are mini snapshots
- Each snapshot has a computed hash, called a commit ID
- Appetite uses this commit ID as an application version



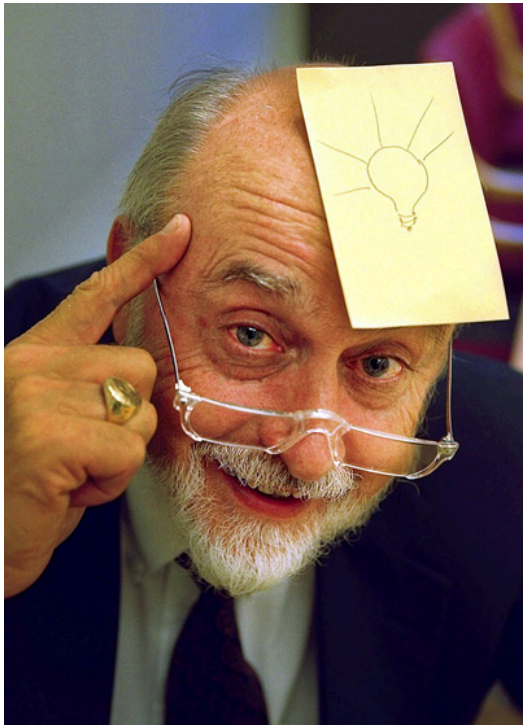
GIT & Splunk Issues

- How do you deploy an application to a host from the repo?
- How do you structure a repo efficiently without duplicating applications?
- How do you track application versions per host?



GIT Repo & Appetite

- Only Two Directories
  - config/
    - deploymenthods.conf
    - manifest.csv
  - apps/
    - Splunk_TA_nix/
    - Splunk_TA_windows/
    - TA_appetite/
    - …



Manifest.csv

- Defines:
  - Applications and specific commitID version
  - Server(s) where app is deployed
  - Method of how it will get there

CommitID	Environment	Application	DeploymentMethod	Whitelist	Blacklist	Comments
c241c6e	PROD	Splunk_TA_nix	ClusterMaster	acme-master.*p$		Unix TA to Prod
1b17b67	DEV	Splunk_TA_nix	Standalone	acme-.*d$	acme-idx.*d$	Unix TA to Dev

- Appetite runs when the commit ID of the manifest changes.

Deploymentmethods.conf

- This file describes the different methods of how applications are deployed within the environment.

[DeploymentServer]

path: "etc/deployment-apps/"

delete_first: true

update_method: "copy"

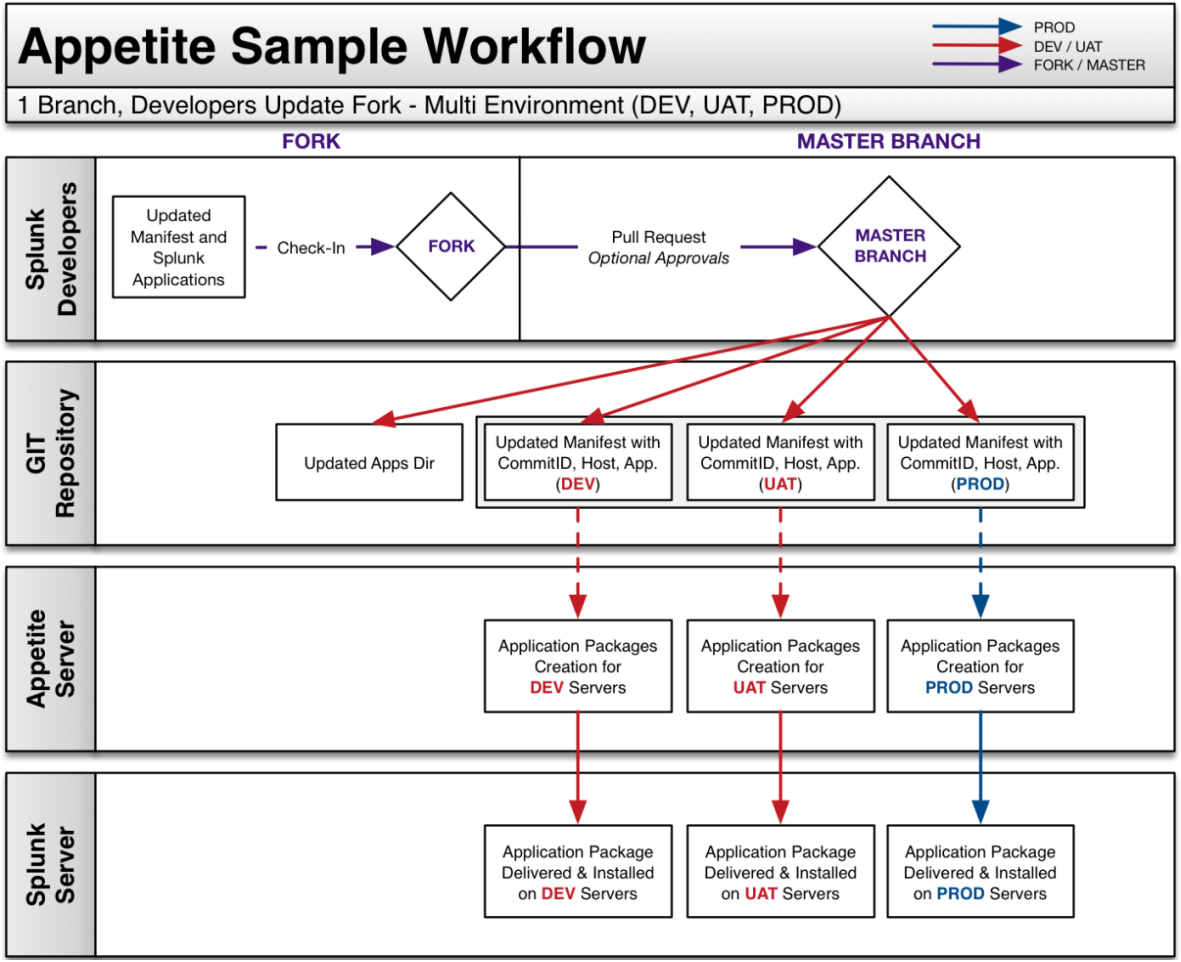
command1: "reload deploy-server --answer-yes”

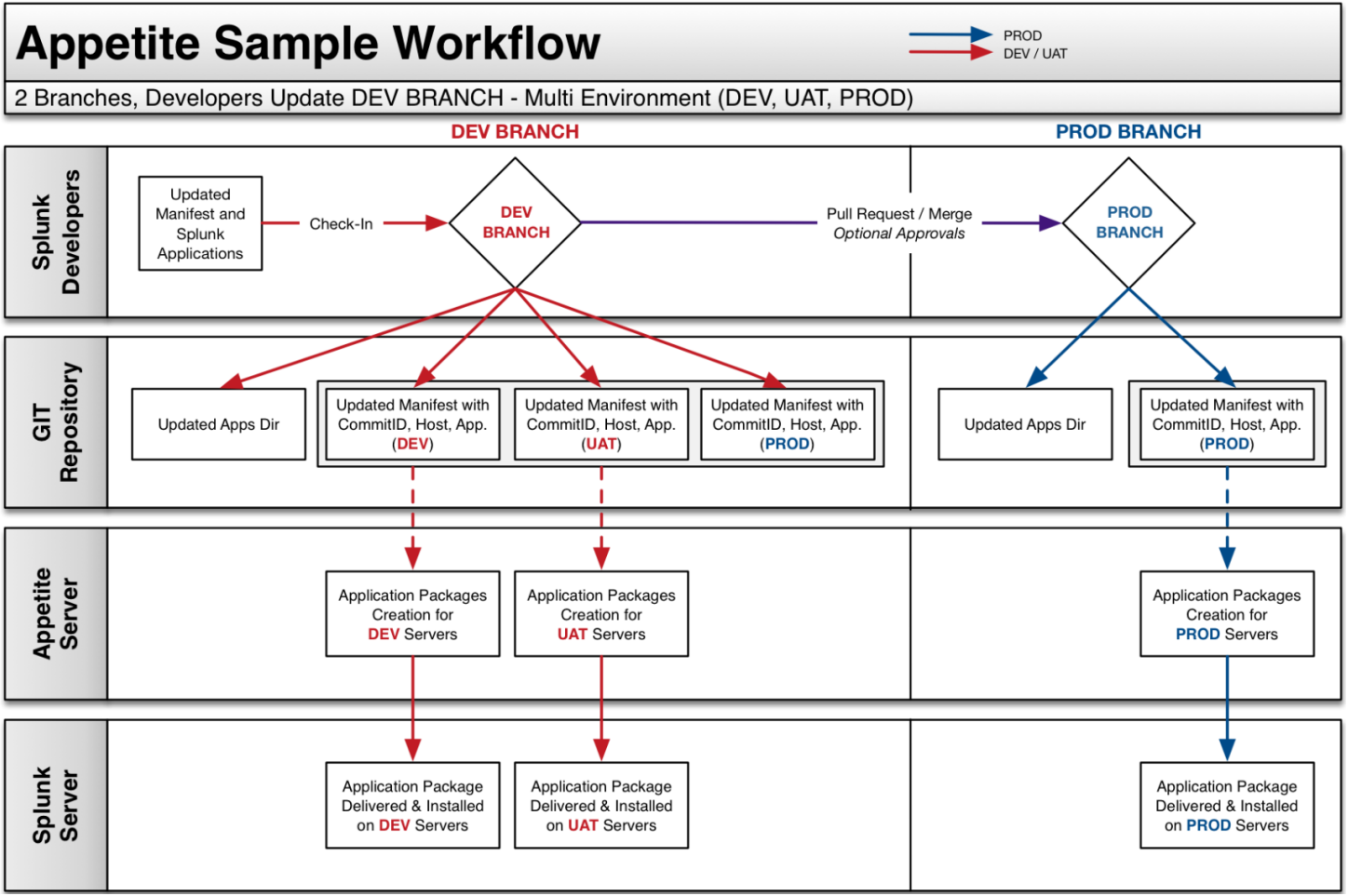
Deploymentmethods.conf & Lookups

- Major pain point!!
- How do you not clobber modified lookups on Search Heads?
- install_ignore = <directory|file list>
  - List of directories or file names which to ignore when being deployed.
  - Appetite will package the application as if these directories or files never existed.
  - IMPORTANT: It is wise to include the lookups directory, to not overwrite lookups changed locally on the server!
- install_inclusion_file = <file_name>
  - Extra file in app root, which lists lookups to include in the application

GIT Workflow

- Highly customizable
- Can set up multiple branches to send to different Splunk environments
  - DEV branch → DEV environment
  - PROD branch → PROD environment
- Can set up one branch to handle all environments
  - Appetite will only send to the servers specified in the manifest
- Configure pull request approvals between branches and/or forks





Appetite Overview

- How does it work?
  - Monitors manifest changes
  - Compares application versions
  - Creates host specific install packages
  - Installs, upgrades, or deletes
  - Modifies Splunk application
- What does it need?
  - Git
  - Python
  - SSH access to Splunk instances

Bon Appétit – Splunk App

- Accompanying Splunk App which audits *everything!*
- Able to track all deployment processes from start to finish

Menu – Manifest Editor

- Makes editing the manifest.csv easier with a UI
- We created a problem:
  - Manifest became unruly
  - Large number of apps
  - Tracking commit IDs
  - Duplicate applications
  - Human errors

Demos

Takeaways & Final Thoughts

- Understand our use case for developing a new system
  - Is this right for your environment?
- Comprehend the high level overview
  - Visualize how it works through our demonstration
- Assess if this is something your company could use
  - Do you have the right resources to implement this solution?
- This is not Splunk supported!
  - Hopefully Splunk comes out with one soon… or bakes this one in? ☺

THANK YOU

splunk>