SUGARCRM

JENKINS AND CONTINUOUS INTEGRATION

Ray Sennewald

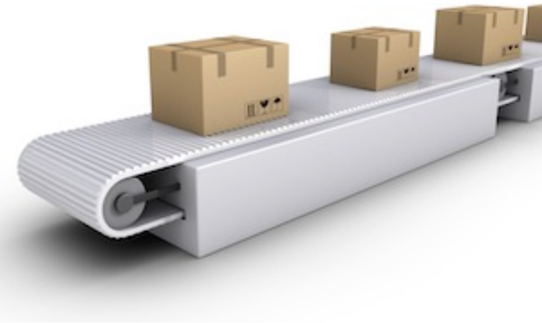
Engineering Tools Lead

2015 SUGARCON

CONNECTING i2i

Continuous Integration

- Software Development Methodology
- Run automated builds on every commit (pull request or commit to feature branch)
- Run all available tests (unit, integration, functional, web services api, etc.)
- Automate all the things
- Maintain a “green” (passing) set of tests and builds
- Notifications should go to the responsible parties when a build fails
- All of your automation scripts should reside in source control (git)



Things to Eliminate

- Manual interactions with the file system, database, etc.
- Building through an IDE
- Manual machine or processing management

Introducing Jenkins

- Jenkins is the swiss army knife of Continuous Integration Applications and is what we use at Sugar. Other options include Travis CI, Bamboo, etc but they aren’t as flexible or provide as much functionality as what we can get from Jenkins.
- Jenkins has countless plugins which can extend it’s functionality. More than likely, someone has already extended Jenkins to do what you need!
- It integrates very well with most of the popular technology that the DevOps community is familiar with such as : Puppet, Docker, Vagrant, AWS/EC2, Git/Github.

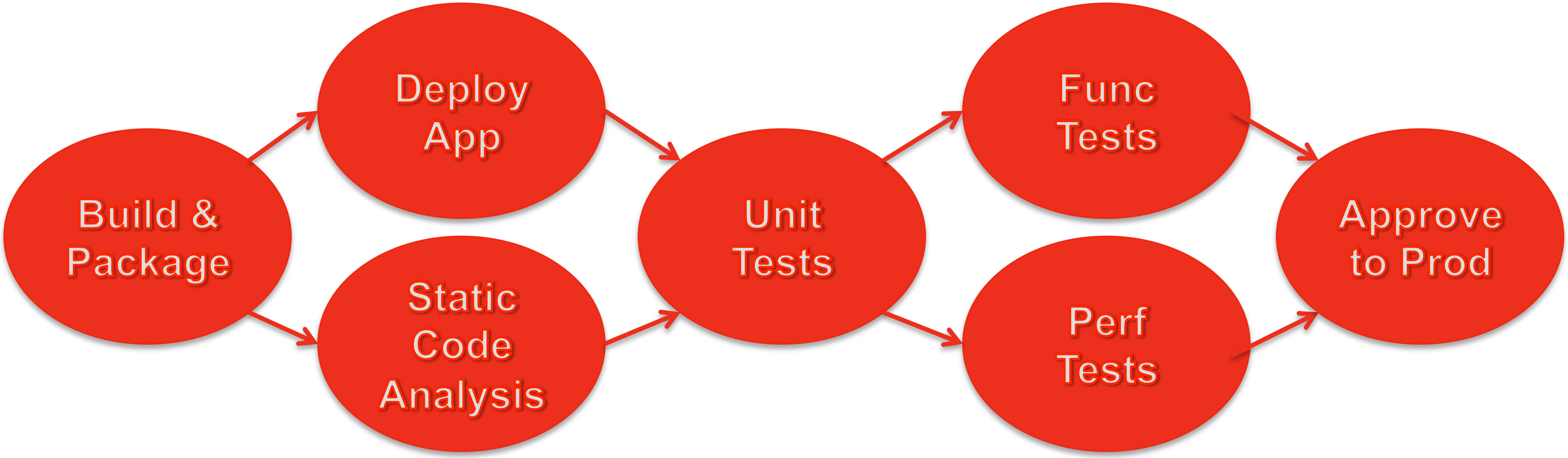
Basic Jenkins

- Git Plugin: Allows you to have a job which will trigger via hook or poll.
- Github Pull Request Builder Plugin: Allows you to have a job which triggers a build for all pull requests and attaches status to commit.
- EC2 Plugin: Allows you to create a set of machine templates based on your AMIs to be used for any jobs.
- Job DSL Plugin: Allows you to write job templates which can be used to create jobs from. (Makes it easy to setup jobs for multiple branches).

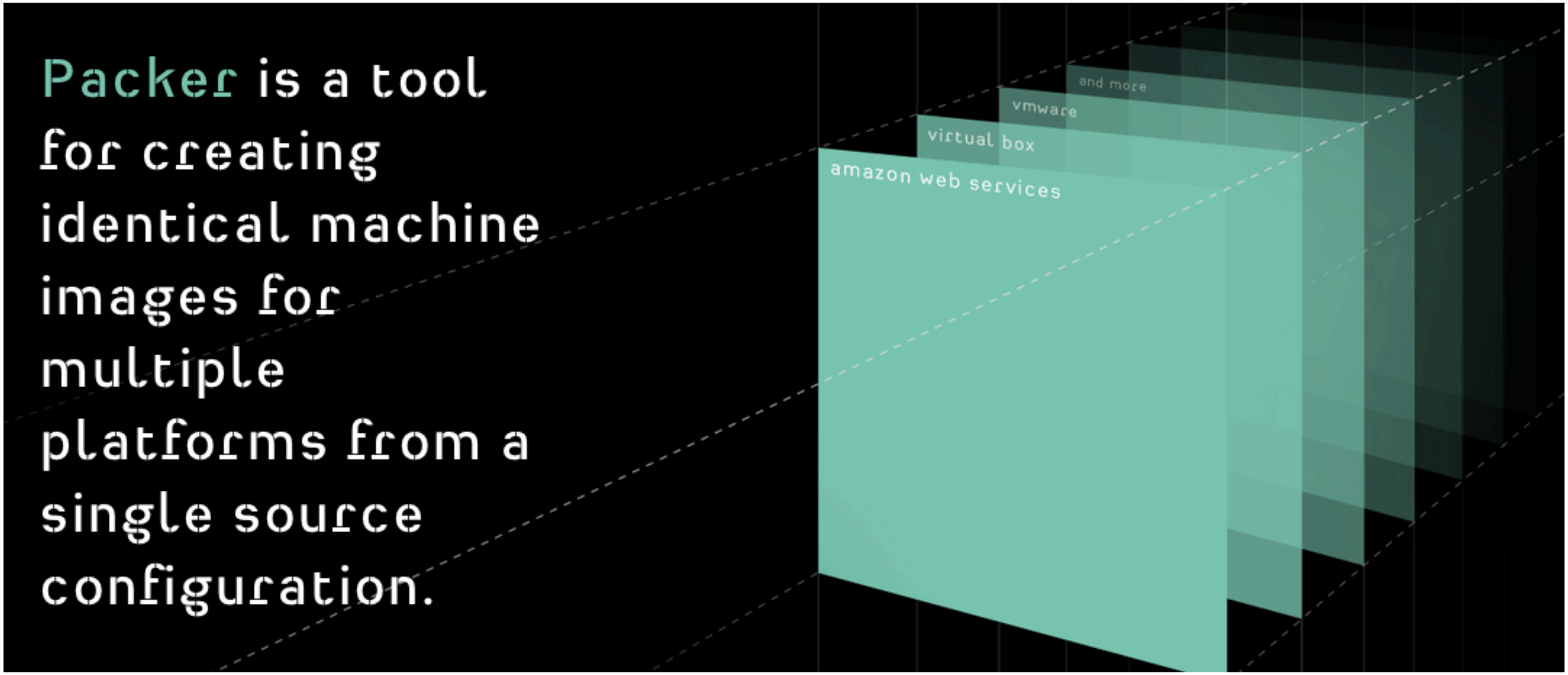
Setting up some Jobs

Demo

Simple Pipeline

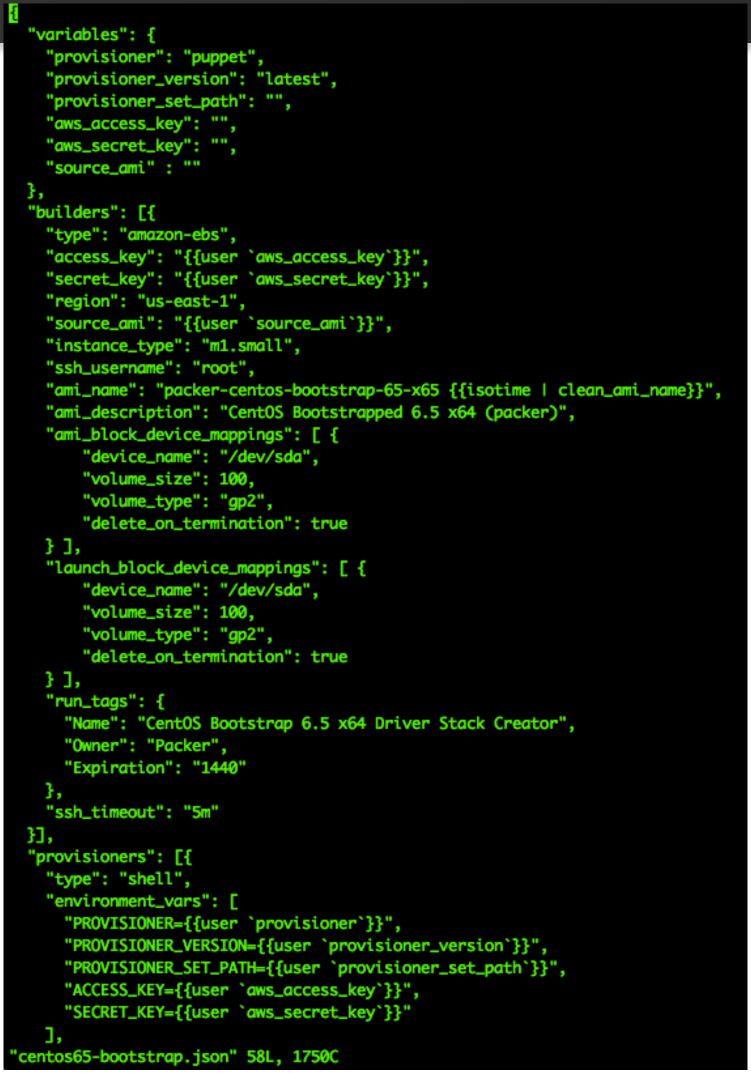


Packer



Packer is a tool for creating identical machine images for multiple platforms from a single source configuration.

Packer Config

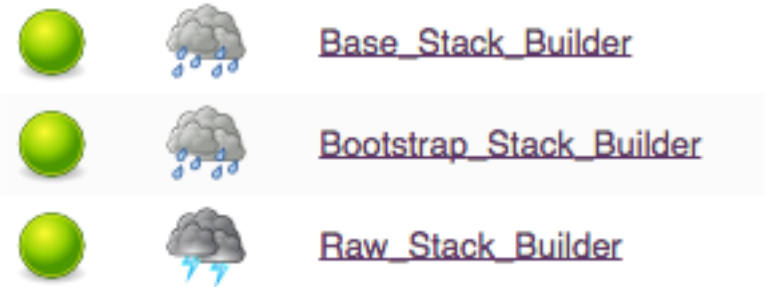


```
{
  "variables": {
    "provisioner": "puppet",
    "provisioner_version": "latest",
    "provisioner_set_path": "",
    "aws_access_key": "",
    "aws_secret_key": "",
    "source_ami" : ""
  },
  "builders": [{
    "type": "amazon-ebs",
    "access_key": "{{user `aws_access_key`}}",
    "secret_key": "{{user `aws_secret_key`}}",
    "region": "us-east-1",
    "source_ami": "{{user `source_ami`}}",
    "instance_type": "m1.small",
    "ssh_username": "root",
    "ami_name": "packer-centos-bootstrap-65-x65 {{isotime | clean_ami_name}}",
    "ami_description": "CentOS Bootstrapped 6.5 x64 (packer)",
    "ami_block_device_mappings": [ {
        "device_name": "/dev/sda",
        "volume_size": 100,
        "volume_type": "gp2",
        "delete_on_termination": true
    } ],
    "launch_block_device_mappings": [ {
        "device_name": "/dev/sda",
        "volume_size": 100,
        "volume_type": "gp2",
        "delete_on_termination": true
    } ],
    "run_tags": {
      "Name": "CentOS Bootstrap 6.5 x64 Driver Stack Creator",
      "Owner": "Packer",
      "Expiration": "1440"
    },
    "ssh_timeout": "5m"
  }],
  "provisioners": [{
    "type": "shell",
    "environment_vars": [
      "PROVISIONER={{user `provisioner`}}",
      "PROVISIONER_VERSION={{user `provisioner_version`}}",
      "PROVISIONER_SET_PATH={{user `provisioner_set_path`}}",
      "ACCESS_KEY={{user `aws_access_key`}}",
      "SECRET_KEY={{user `aws_secret_key`}}"
    ],
"centos65-bootstrap.json" 58L, 1750C
```

- Packer configs are just json files.
- Can build machine images to many target platforms (Amazon, VMware, VirtualBox).
- Can execute shell scripts to help bootstrap your server.

Workflow

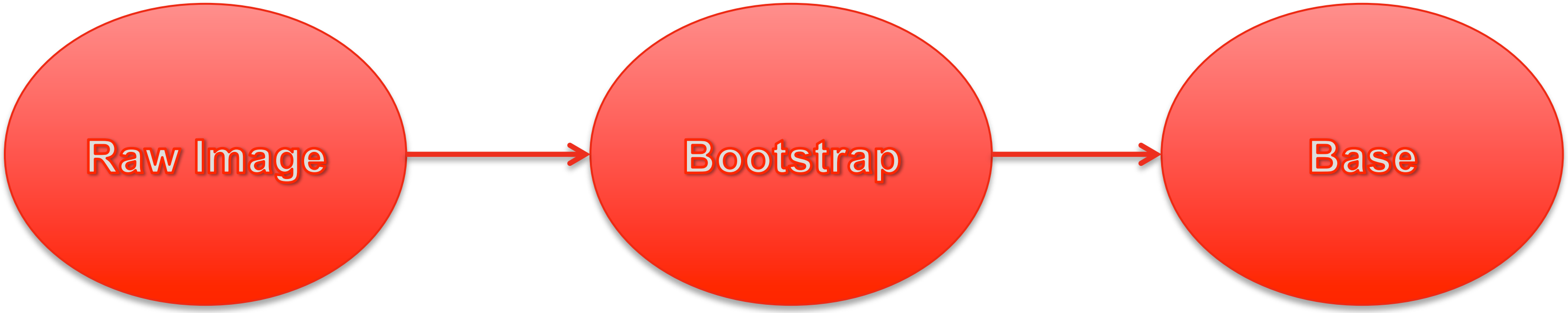
- Use Jenkins to Automate Building the Machines.
- Have whole workflow automated.



● Base_Stack_Builder

● Bootstrap_Stack_Builder

● Raw_Stack_Builder



Questions?

SUGARCRM

THANK YOU

2015
SUGARCON
CONNECTING **i2i**