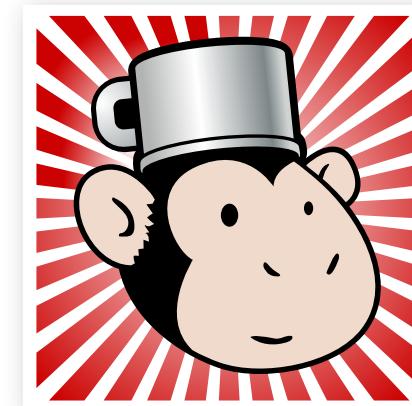


JUDCon

JBoss Users & Developers Conference

Boston:2011



BoxGrinder

Marek Goldmann

Agenda

- whoami
- Some background and terminology
- BoxGrinder
 - Appliance definition files
 - Architecture overview
 - Build process
 - Writing a plugin
- Small demo

Who's Marek?

- 
- JBoss Developer
 - Cloud-related projects: lead of  **BoxGrinder**
- Part of **project:odd**
- Electronic music lover



Some terminology

Appliance is a preconfigured disk image
(virtual machine) with operating system
and all required applications to do
specific job

Appliance examples with tasks

- **Database**
 - Storing data
- **Front-end**
 - Load balancing
- **Back-end**
 - Actual servers

Bake vs. Fry

Bake: Produce a complete virtual machine offline, before first use.

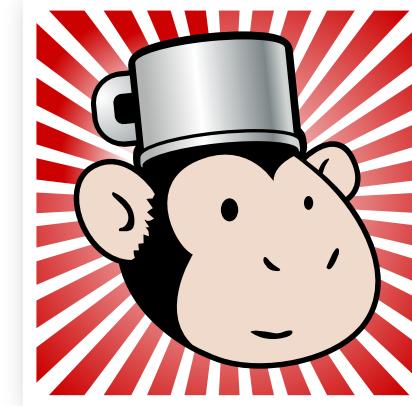
Fry: Produce a complete virtual machine by booting a basic VM and then applying configuration.

Bake!

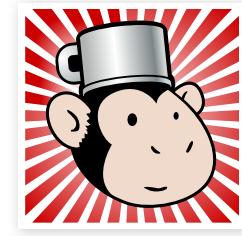
We think **baking** is The Right Way, especially for developers simply looking for reliable platforms.

Bake, then fry

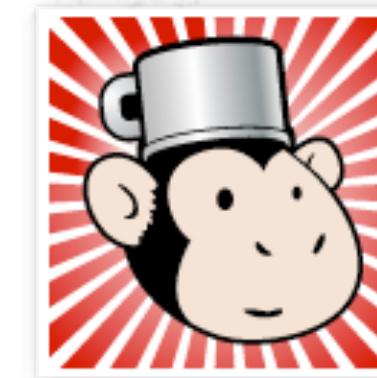
If you **bake** an image you can **fry** it
then later too! Baked image can be
your **start point**.



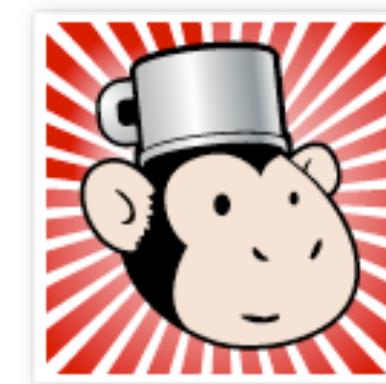
BoxGrinder



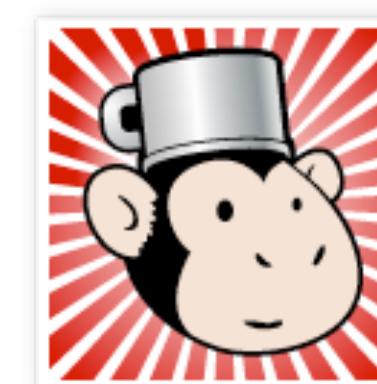
BoxGrinder is a family of tools to grind
out **appliances** for various **platforms**



BoxGrinder
Build

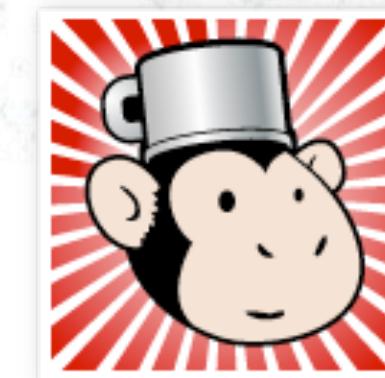


BoxGrinder
REST



BoxGrinder
Studio

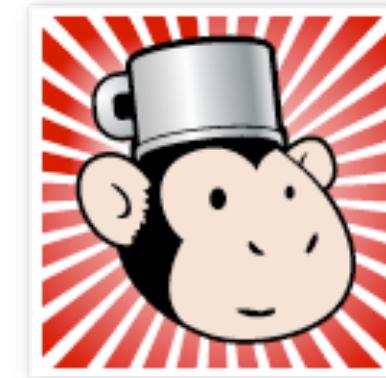
Current status



BoxGrinder
Build



stable



BoxGrinder
REST



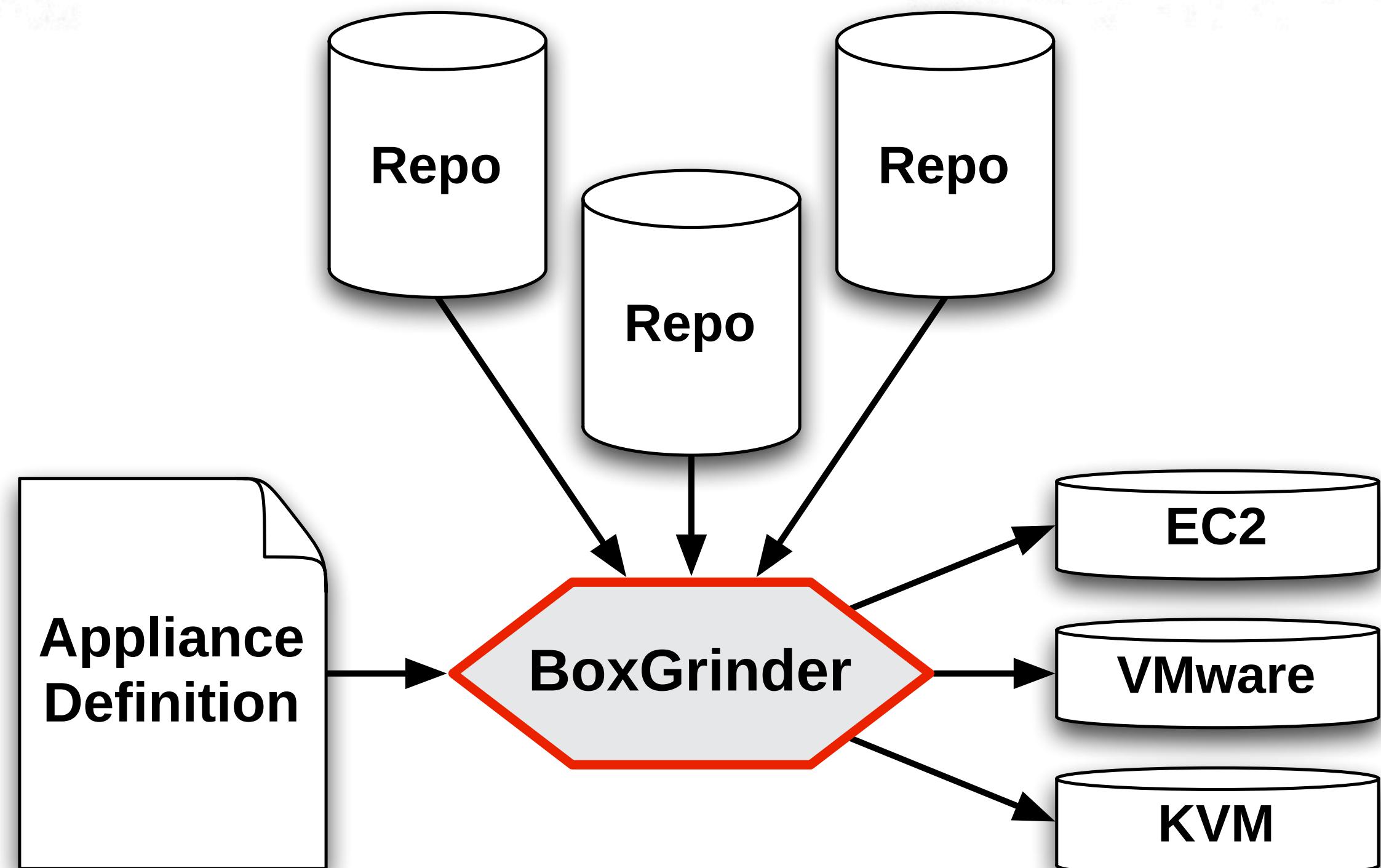
development



BoxGrinder
Studio



planning



Closer look at appliance definition file

Appliance definition, huh?

- Plain text file – **YAML** format
- Very easy to understand, modify
- Inheritance (mixins)

Appliance example

```
name: back-end
version: 1
release: 1
summary: back-end appliance with JBoss AS 6
hardware:
  memory: 512
  partitions:
    "/":
      size: 2
appliances:
  - fedora-base
packages:
  - jboss-as6
  - jboss-as6-cloud-profiles
  - java-1.6.0-openjdk
...
...
```

General information

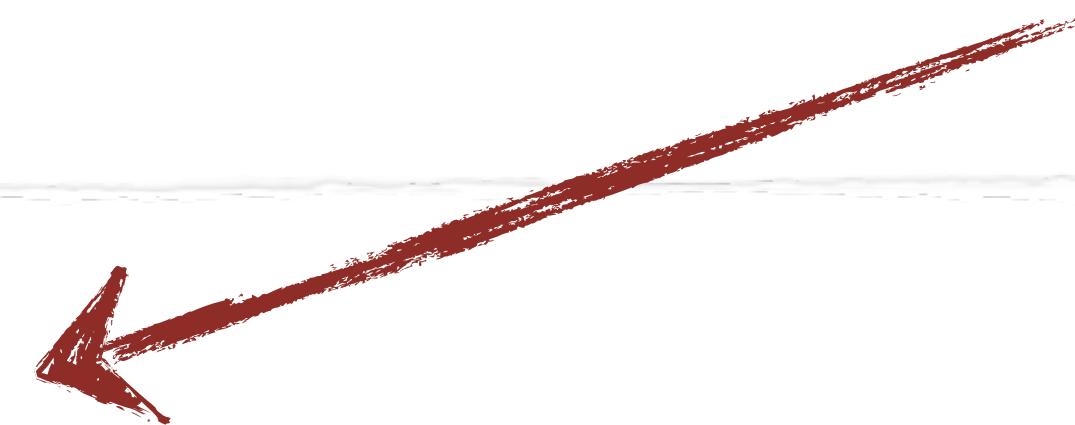
should match the filename: back-end.appl

name: back-end

version: 1

release: 1

summary: back-end appliance with JBoss AS 6



Hardware

```
hardware:  
  memory: 512  
  partitions:  
    "/":  
      size: 2
```

512MB

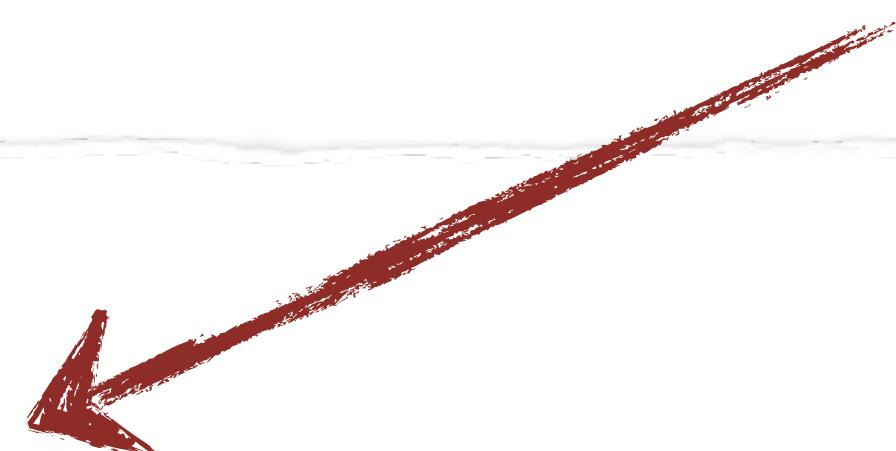
2GB



Appliance Mix-ins

Mixing in **fedora-base.appl**

appliances:
- fedora-base



back-end.appl

```
name: back-end
version: 1
release: 1
summary: back-end appliance with JBoss AS 6
hardware:
    memory: 512
    partitions:
        "/":
            size: 2
appliances:
    - fedora-base
packages:
    - jboss-as6
    - jboss-as6-cloud-profiles
    - java-1.6.0-openjdk
...
...
```

fedora-base.appl

```
name: fedora-base
summary: Basic Fedora OS
os:
  name: fedora
  version: 14
hardware:
  memory: 256
  partitions:
    "/":
      size: 1
packages:
  - @core
  - openssh-server
  - openssh-clients
  - wget
```

Appliance Mix-ins

back-end.appl

overrides

fedora-base.appl

```
hardware:  
  memory: 512  
  partitions:  
    "/":  
      size: 2
```

```
hardware:  
  memory: 256  
  partitions:  
    "/":  
      size: 1
```

Appliance content

packages:

- jboss-as6
- jboss-as6-cloud-profiles
- java-1.6.0-openjdk

Appliance content

packages:

- jboss-as6
- jboss-as6-cloud-profiles
- java-1.6.0-openjdk

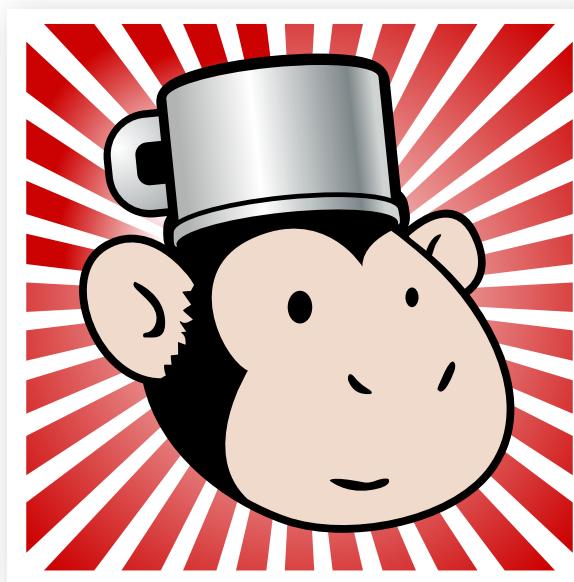
Plus everything from
fedora-base.appl

There is a lot more!

- Additional sections
 - **repos**
 - ephemeral repos
 - **post**
 - What should be done **after** you build your appliance
 - Different commands for different platform
 - Using **libguestfs**
- Learn more!

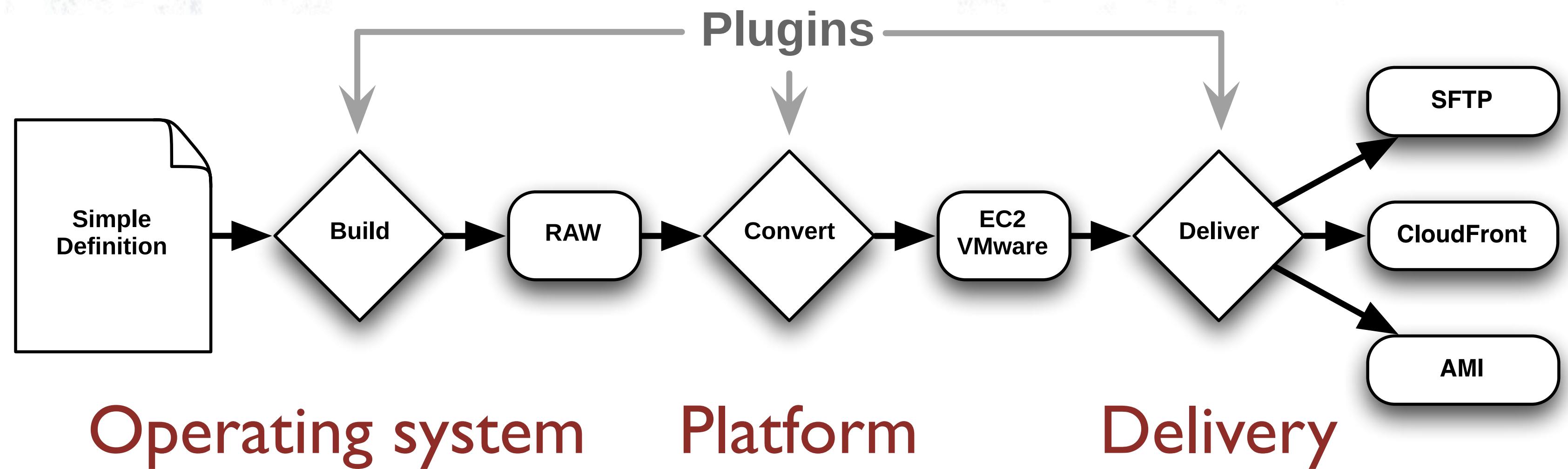


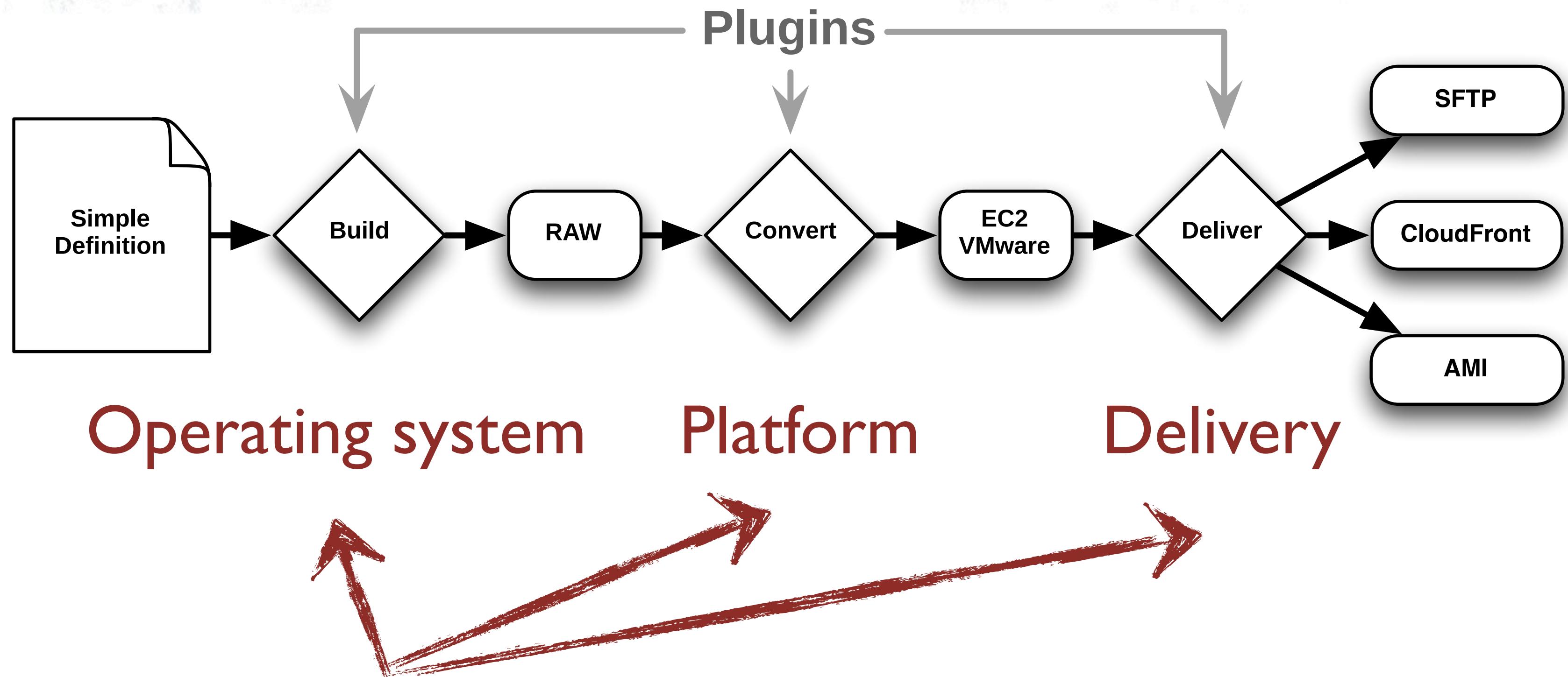
<http://boxgrinder.org/tutorials/appliance-definition/>



BoxGrinder Build

BoxGrinder Build architecture





<http://boxgrinder.org/tutorials/how-to-write-a-plugin-for-boxgrinder-build/>

Plugin skeleton

```
require 'boxgrinder-build/plugins/base-plugin'

class YourPlugin < BoxGrinder::BasePlugin
  def execute
    # PLACE YOUR CODE HERE
  end
end
```

Plugin registration

```
require 'boxgrinder-build/managers/plugin-manager'  
require 'xyz-your-plugin/your-plugin'  
  
plugin :class => YourPlugin,  
       :type => :platform,  
       :name => :mycloud,  
       :full_name => "MyCloud"
```

How to **install** BoxGrinder Build



yum install rubygem-boxgrinder-build

Meta appliance

A preconfigured appliance to **build other appliances** using BoxGrinder

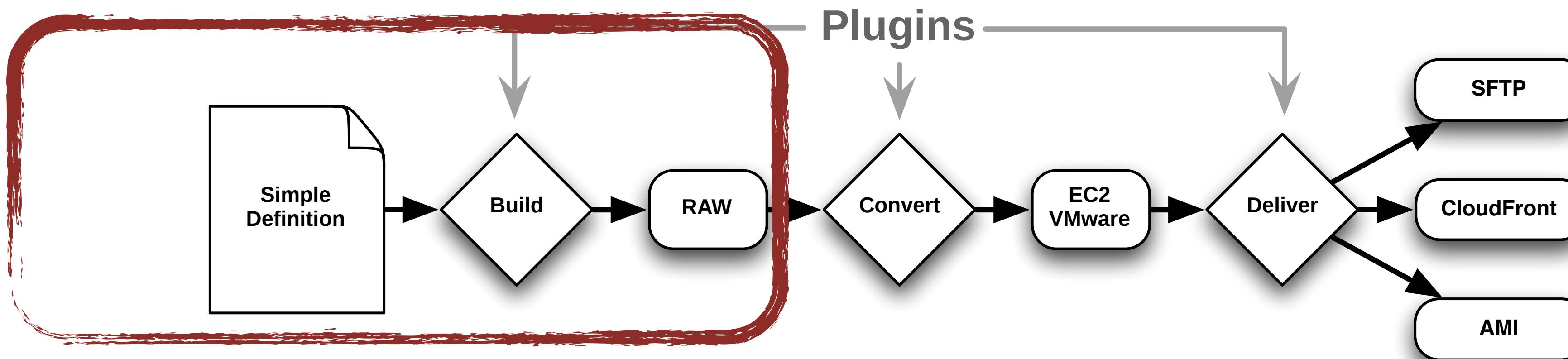
- Easy to jump in
- Available for different platforms: Xen, KVM, EC2, VMware
- **Best way** to build EC2 appliances

<http://boxgrinder.org/download/boxgrinder-build-meta-appliance/>

Demo: build a simple appliance

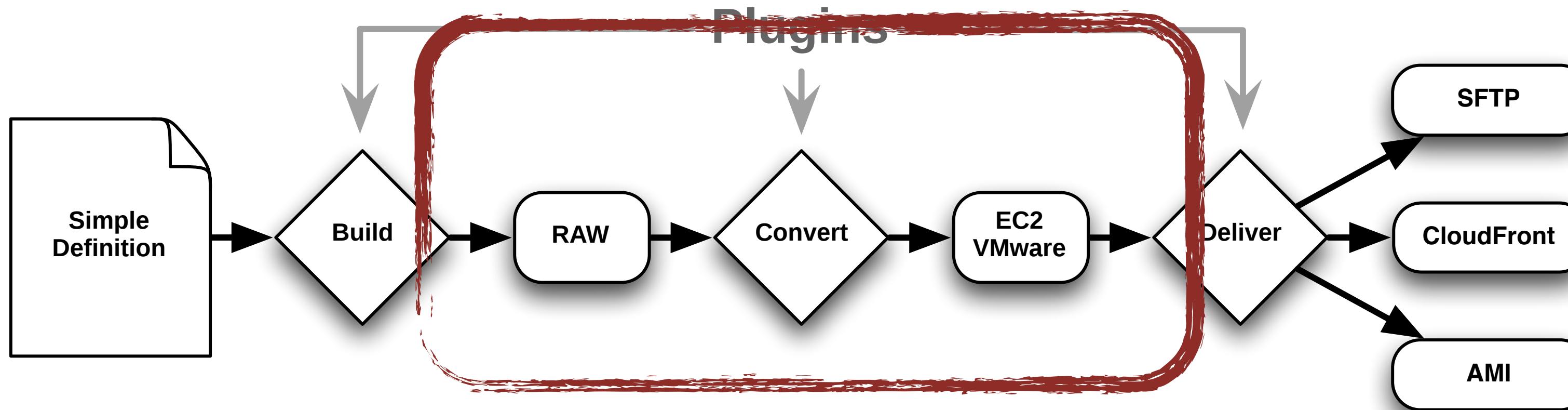
convert and deliver
~~Demo: build~~ a simple appliance

Step 1: create base image



`boxgrinder-build f14-jeos.appl`

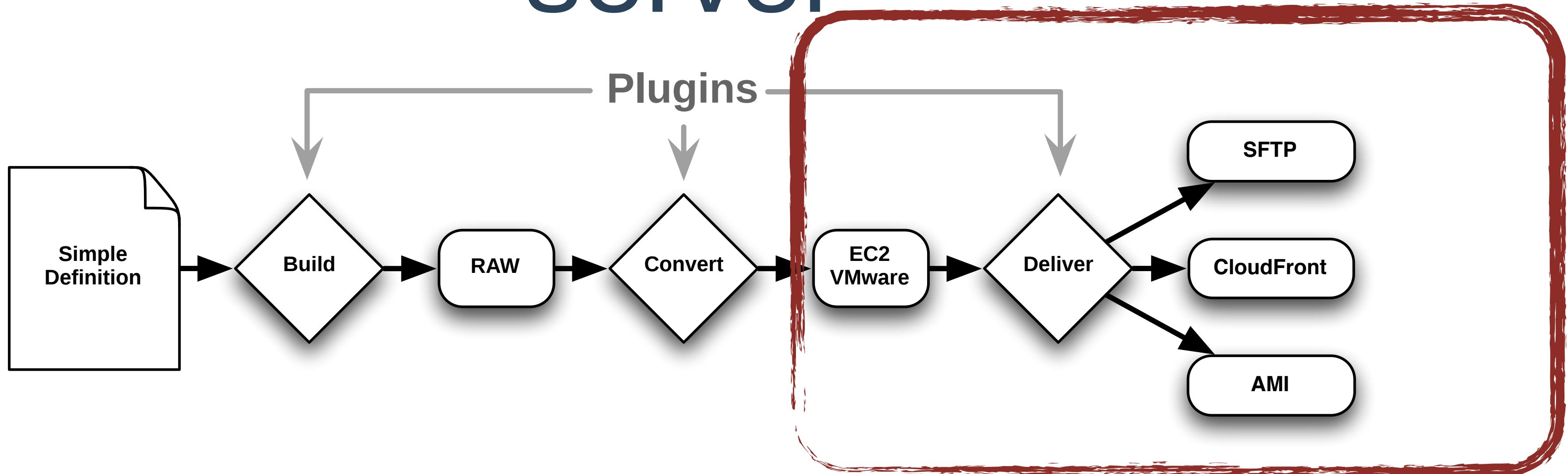
Step 2: convert it to VMware type



```
boxgrinder-build f14-jeos.appl -p vmware  
-p ec2
```

...

Step 3: deliver it to a SFTP server



```
boxgrinder-build f14-jeos.appl -p vmware -d sftp  
-d ebs  
-d s3
```

Of course you can run the command
just **once** with same result!

```
boxgrinder-build f14-jeos.appl -p vmware -d sftp
```

What's **hot**?

BoxGrinder Build features

- Supported OSes: **Fedora, CentOS, RHEL**
- Supported platforms: **EC2** (S3-based and EBS-based too!), **KVM, VMware, VirtualBox**
- Many delivery options: **local, SFTP, S3 or CloudFront** as tarred image, **AMI**
- Many public clouds supported: EC2, ElasticHosts, Serverlove, Open Hosting, SKALI Cloud, CloudSigma (coming)

BoxGrinder Build features

- **Cross-arch builds:** producing i386 images on x86_64 hosts
- Caching downloaded resources (RPM's)
- **Pretty fast** – from .appl to registered AMI: **20 minutes** (on EC2, using meta-appliance)

Notes

- If you're building AMI's – **do it on EC2** – this will save your time (uploading to S3 from your local machine isn't fun...)
- Building **EBS-based AMI's requires** to run BoxGrinder on EC2

Questions?

<http://github.com/boxgrinder/>

<http://boxgrinder.org/>

<http://boxgrinder.org/blog/>

IRC: **#boxgrinder**



@boxgrinder
@marekgoldmann

Thanks!