MetaCentrum

# A Cloud Image Factory

## Managing Virtual Appliance Lifecycle in IaaS and PaaS Clouds

Michal Kimle, Ľubomír Košarišťan, Boris Parák, Zdeněk Šustr

CESNET

18 March 2016

20 1996–2016 CESNET e-infrastructure for science, research and education

ISGC 2016, Taipei, Taiwan

# Motivation I.

**Trends in IaaS and PaaS clouds**

- users can easily use their favourite distribution/appliance
- users can choose from variety of distributions/appliances
- users expect their distributions/appliances to be up-to-date and secure
- advanced users expect possibility to create and use their own distribution/appliance

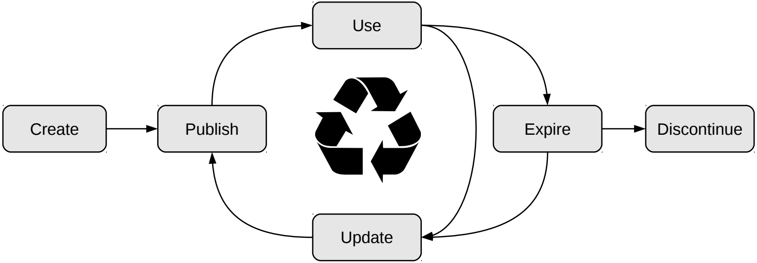
# Motivation II.

**Pros of proper (automatic) virtual appliance life-cycle management**

- following the aforementioned trends ⇒ happy users
- providing useful jumping-off points for new users
- isolating users from technical details of the infrastructure
- providers have better control over appliance (updates & security)

# Motivation III.

## Cons? (or What does it take?)



Virtual Appliance Life-cycle

# Motivation IV.

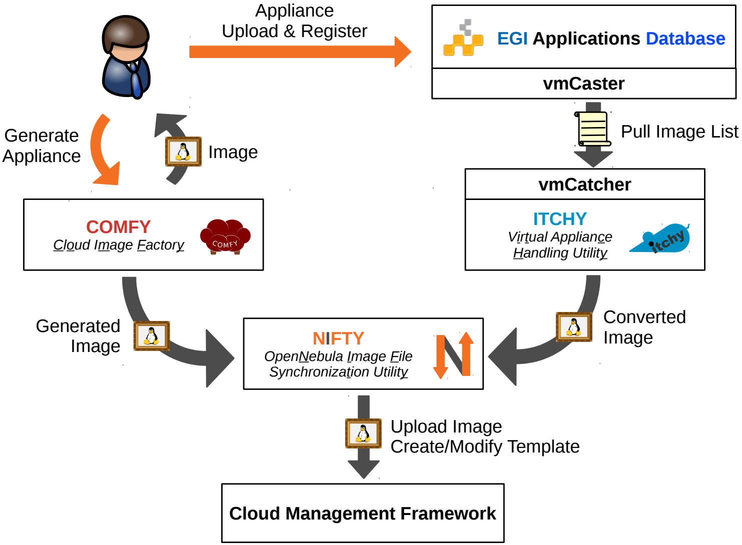
## Cons? (or What does it take?)

- Generating Appliances
  - minimalistic OS installation
  - basic configuration for remote access
  - contextualization support (e.g., *cloud-init*)
  - additional software specific for the appliance
- Distributing Appliances
  - a place to publish appliances
  - a way to update/expire existing appliances
  - some communities already have solutions in place
    - The *EGI* Application Database – `https://appdb.egi.eu/`
    - *HEPiX* vmCaster – `https://goo.gl/yBOec9`
    - *HEPiX* vmCatcher – `https://goo.gl/1qMGlW`

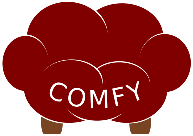
# Solution

We have the right tools!

# The Big Picture



# COMFY I.



*Cloud Image Factory*

**What does it do?**

- creates virtual appliance images
- supports common distributions
- ships with reasonable defaults
- minimal configuration
- useful both for users and providers

# COMFY II.

*Cloud Image Factory*

**How does it work?**

- image based on description of appliance properties
  - image format, disk layout
  - Linux distribution
  - customization & configuration
- generating base appliance
  - *packer.io* + *QEMU*, *VirtualBox*
- running provisioning (shell scripts, chef, puppet, ...)
- generating appliance metadata (JSON-formatted descriptor)

# ITCHY

*Virtual Appliance*
*Handling Utility*

### What does it do?

- handles appliance sanitation and image conversion
- solution specific for EGI Fedcloud infrastructure

### How does it work?

- The EGI Application Database → HEPiX vmCaster (signature)
- HEPiX vmCaster → HEPiX vmCatcher (signature verification)
- vmCatcher event handling (new appliance, update, expiration, . . . )
- unification of appliance image formats (OVA, qcow2, vmdk, vdi, . . . )
- generating appliance metadata (JSON-formatted descriptor)

# NIFTY

*OpenNebula Image File Synchronization Utility*

### What does it do?

- uploads and registers virtual appliances
- currently supporting OpenNebula cloud platform
- modular design

### How does it work?

- reading metadata (JSON-formatted descriptor)
- appliance event handling (new, updated, expired, . . . )
- registering images and creating templates
- handling expiration & delayed clean-up

# Roadmap

Current progress

- testing & debugging
- first stable release of all components
- large-scale deployment in the EGI Federated Cloud
- appliances generated by COMFY already in production in MetaCloud

Future development

- support for other image distribution platforms (OpenNebula Marketplace, EGI AppDB)
- appliance provisioning via puppet recipes
- better support for custom user appliances (via GIT repositories)
- hierarchical structure of appliances (Docker-like)
- more and more ready to use appliances

# The End

– That's All! –

. . .

# Contact Info

Do you have any questions?

- ask **NOW!**
- ask us directly
  - kimle@cesnet.cz
  - parak@cesnet.cz
  - lubomir.kosaristan@gmail.com
  - sustr4@cesnet.cz
- send your questions to cloud@metacentrum.cz

Something to explore:

- https://github.com/CESNET/comfy
- https://github.com/CESNET/itchy
- https://github.com/CESNET/nifty