

---

# Gestion des Appliances

M. AIRAJ  
Ecole Informatique IN2P3  
2014-2015

# Appliances pour logiciels et infrastructures différentes?

## □ Logiciels :



OpenNebula.org



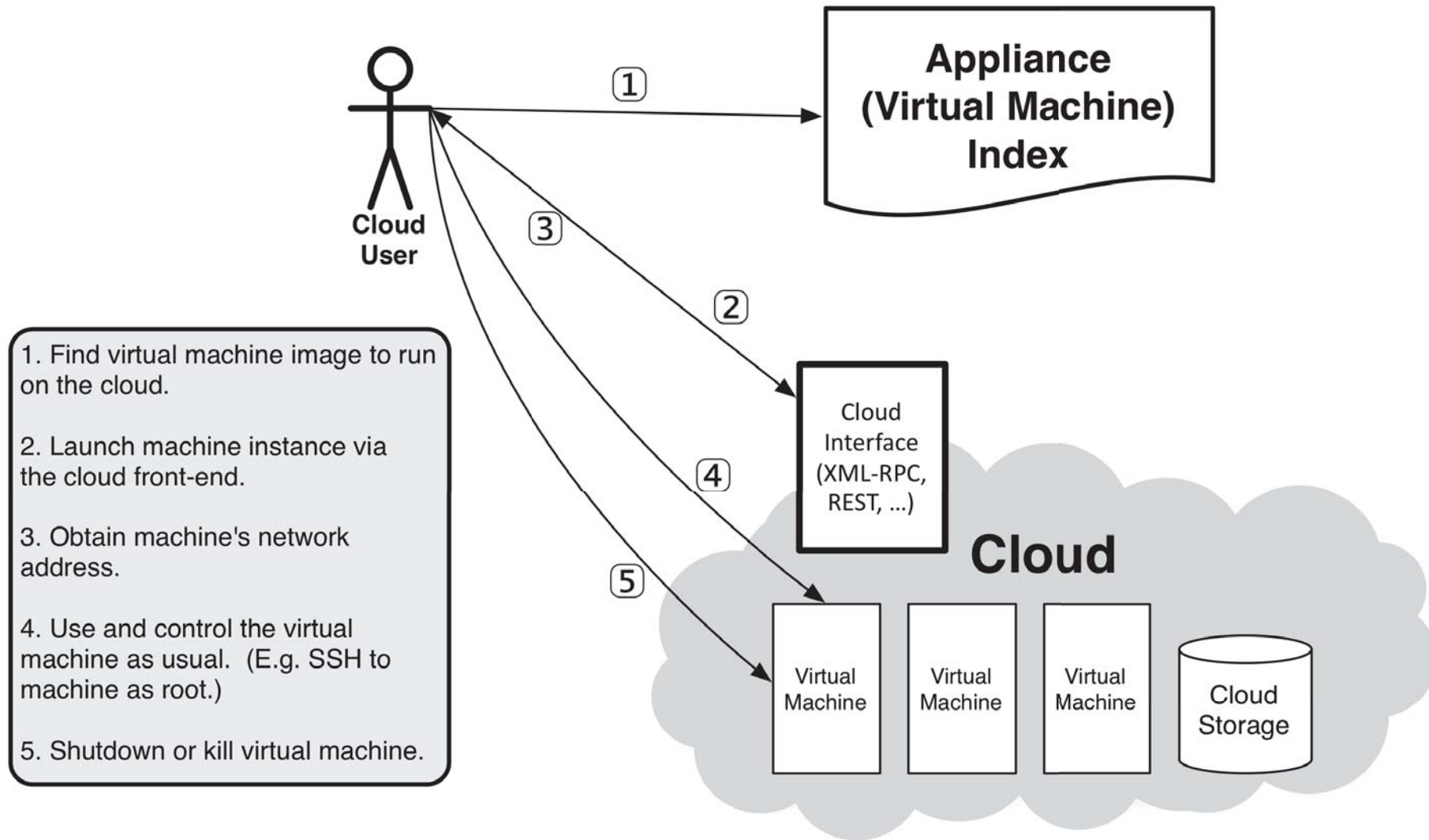
EUCALYPTUS

## □ Infrastructures :



ElasticHosts  
Flexible servers in the cloud

# Déploiement d'une appliance



# Appliance : Définition

## ■ Définition

- Image générique avec un OS
- Optionnellement contient des services préinstallés/configurés

## ■ Bénéfices

- Facilite un accès rapide en utilisant les ressources du cloud
- Le packaging permet d'éviter les complications d'installation

# Appliances : pour que ça marche!

- Pour qu'une appliance soit instanciée correctement sur une infrastructure de Cloud, elle doit vérifier ces deux éléments techniques :
  - Son format doit être supporté par l'infrastructure Cloud
  - Outils/scripts de contextualisation doivent être installés dans l'appliance et supportés par l'infrastructure Cloud.

# Appliances : Questions?

- Existe-t-il un format d'appliance standard?
- Existe-t-il un outil de contextualisation standard?
- **Encore plus simple**, les infrastructures de Cloud ne peuvent-elles pas faire les conversions et adaptations nécessaires?

# Format des images des appliances

■ Il existe plusieurs formats des images des appliances :

- RAW,
- FVD,
- QCOW2,
- VirtualBox VDI,
- VMware VMDK,
- Microsoft VHD,
- Ovf (standard),
- ...

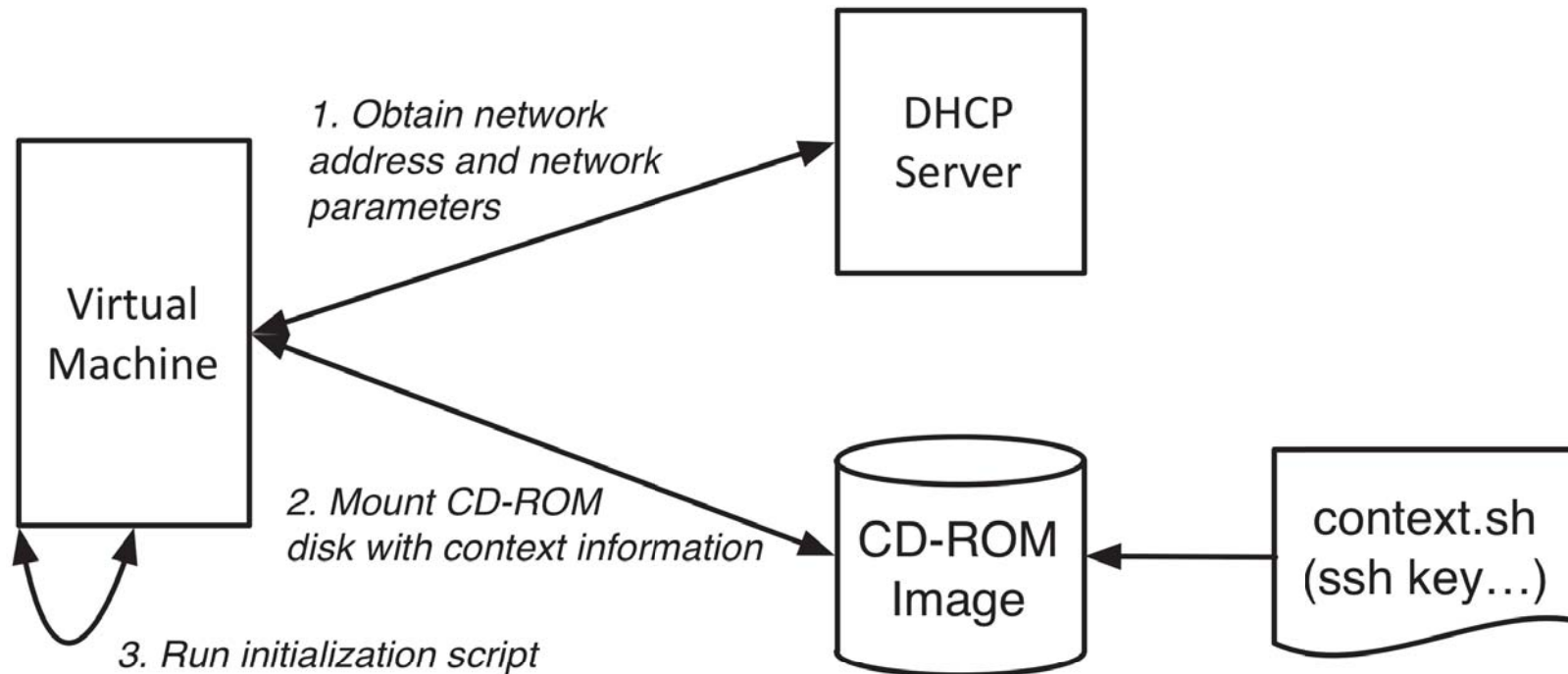
# Contextualisation d'une appliance

- Contextualisation est le processus qui permet à une VM instanciée de connaître son environnement cloud « contexte », permettant à la VM de se configurer et démarrer ses services correctement.
  - par exemple le nom d'hôte, adresse IP, ssh / authorized\_keys, les services en cours d'exécution, ...)
- On distingue deux phases principales :
  - Obtenir des données de contexte dans la VM
  - L'utilisation de ces données pour la personnalisation de la VM



# Contextualisation : Exemple

- Contextualisation d'une VM dans StratusLab, méthode CDROM :
  - Connexion réseau (DHCP)
  - Authentification/Autorisation (clés SSH)
  - Configuration de services (e.x. Certificats Grille)



# Contextualisation d'une appliance

- Sources de données possibles pour une VM:
  - serveur de métadonnées (OpenStack)
  - système de fichiers attachés/disque (OpenStack / StratusLab),

# Contextualisation d'une appliance : standard de contextualisation?

## CloudInit

Le mécanisme **CloudInit** est un outil de contextualisation qui sait gérer les deux méthodes : serveur web et disque.

Il est assez simple à mettre en oeuvre.

# Contextualisation d'une appliance

- ▣ **Cloud-init** détecte automatiquement la source des données et contextualise suivant une syntaxe bien définie pour les données.

	<i>OpenStack</i>	<i>StratusLab</i>	<i>OpenNebula</i>	<i>AmazonEC2</i>
<i>Disque</i>	✓	✓	✓	✓
<i>Cloud-init</i>	✓	✓	✓	✓

# Appliances : Questions?

- Existe-t-il un format d'appliance standard?

Ovf en est un, mais pas encore largement adopté.

- Existe-t-il un outil de contextualisation standard?

CloudInit est adopté/supporté par plusieurs infrastructures/logiciels de Cloud.  
Probablement un futur standard.

- Les infrastructures de Cloud ne peuvent-elles pas faire les conversions et adaptations nécessaires?

Quelques unes, partiellement.

# Création des appliances

- Une appliance peut être créée par différents moyens :
  - Manuellement
  - Utilisation des « Usines »
  - Outils de création automatisés en ligne de commande
  - ...

# Création manuelle d'appliance à partir de zéro

- Peut se faire par plusieurs méthodes : virtualisation desktop, libvirt ...
- Le moyen le plus simple est d'utiliser une solution de virtualisation desktop.
  - Le résultat peut être convertit au format requis par l'infrastructure Cloud.
- La création manuelle d'appliances :
  - prend beaucoup d'effort et de temps,
  - n'est pas automatique,
  - reste un sujet à erreurs,
- Donc à éviter si on a le choix. Ne le faire si vraiment il n'y a pas autre alternative!
- Le mieux est d'adapter une appliance existante.

## Création des appliances : Les Usines

- Les usines à appliances fournissent des appliances pré-fabriquées prêtes à être déployer.
- Elles peuvent être utilisées telles quelles ou comme appliances de base que l'utilisateur peut personnaliser avec ses propres logiciels et services.
- Exemple d'usine à appliances : Bitnami



## Création des appliances : Outils automatiques

- Un très grand nombre d'outils en ligne de commande permettent l'automatisation du processus de création d'appliances.
- En général, ils suivent un schéma commun de l'installation d'un système d'exploitation de base sur une image disque nouvellement créé, puis la personnalisation de cette image.
- En général permettent aussi de faire des changements incrémentaux aux appliances de base.
- Exemple d'outils en ligne de commande : VMBuilder, VeeWee, BoxGrinder , ...

# Création des appliances : Bonnes pratiques

Service/règle/package	Recommandations
Firewall	désactivé
sécuriser ssh	Arrêt de tous les accès non-clés
Network boot protocol	Dhcp,...
Outils/scripts de contextualisation	Cloud-init, Hepix, ...
udev	<pre>rm -f /lib/udev/rules.d/*net-gen* rm -f /etc/udev/rules.d/*net.rules</pre>
selinux	désactivé
swap	Rajouter le swap disk dans fstab
acpid	Nécessaire pour l'attache et le détache à chaud de volumes
Hostname	Supprimer la référence au hostname d'installation de la configuration réseau.

# Création des appliances : Bonnes pratiques

## En résumé

- Etre sûr que toutes les informations privées ont été supprimées de l'image
- Inclure les outils/scripts de contextualisation
- Informations réseaux sauvegardées dans le cache : adresse mac, adresse ip , ...
  - ➔ Supprimer toutes ces informations
- Arrêter les services pour éviter des trous de sécurité

# Dépôt d'Appliances (Repository)

- Afin d'instancier une VM à partir d'une appliance :
  - Cette appliance doit exister quelque part et est accessible depuis le Cloud.
- Les différentes distributions de Cloud utilisent différents outils pour sauvegarder les appliances :
  - **OpenStack** fournit un service « Glance » pour enregistrer, rechercher et délivrer une appliance.
    - Une appliance peut être sauvegardée dans un simple système de fichier, comme dans swift un système de stockage objet.
    - Les appliances et leurs meta-données sont proposées via l'API Glance
  - **StratusLab** sauvegarde les appliances dans un endroit accessible via le web.
    - Les métadonnées sur ces appliances contiennent l'information sur leur localisation, et sont publiées dans le Marketplace.
    - Le Marketplace propose une simple interface REST, plus un Web Browser .

# Registre d'Appliances

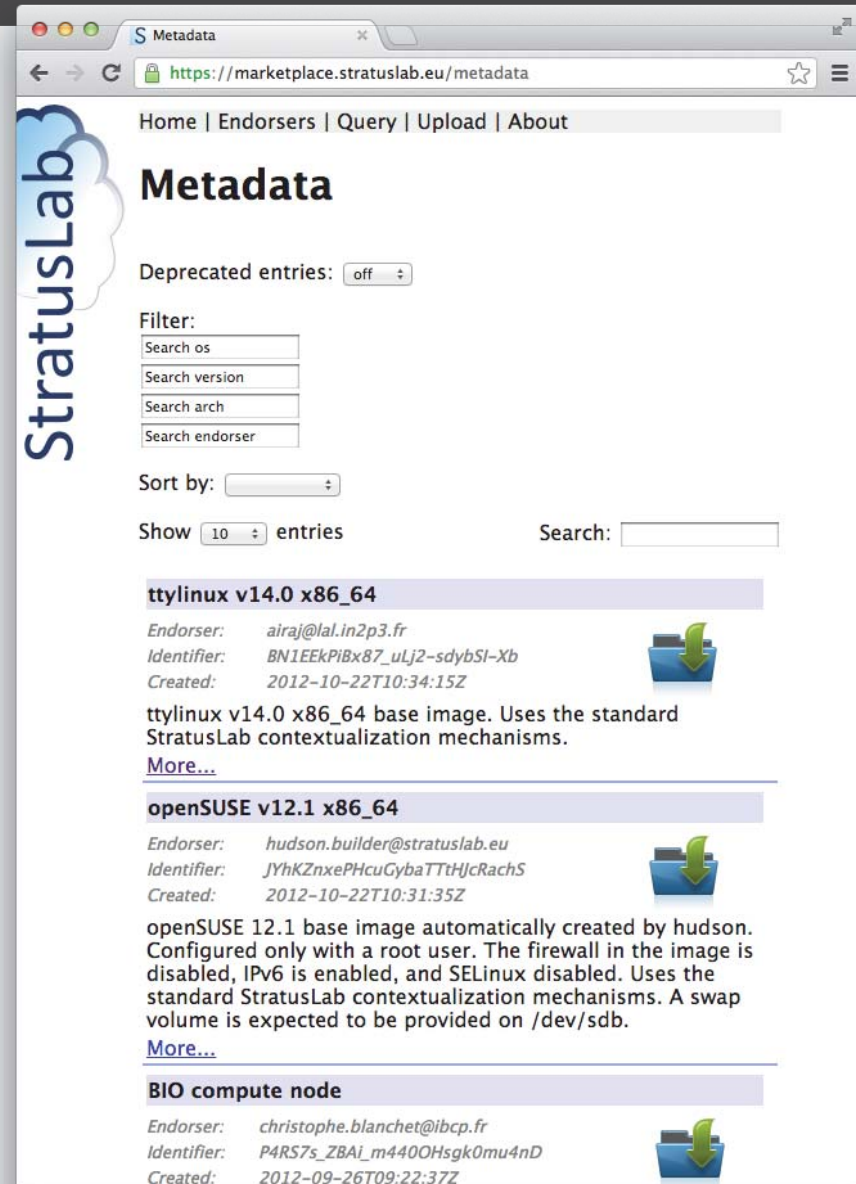
- Un registre d'appliances permet aux utilisateurs de rechercher des appliances existantes sur la base de certains critères.
- Toute distribution de Cloud qui permet aux utilisateurs de partager des appliances doit avoir un registre d'appliances.
  - La plupart des distributions fournissent un tel service, bien que dans beaucoup de cas, il est intégré avec le repository d'appliances...
- Exemple : Le Marketplace de StratusLab
  - Il est au centre du mécanisme de gestion d'appliances dans le Cloud StratusLab.
  - Il contient les métadonnées sur les appliances.
  - Il sert de registre pour les appliances partagées.
  - Pour utiliser et/ou partager une appliance, ses métadonnées doivent être enregistré dans le marketplace.

# StratusLab : Marketplace

*Identifiant est le fingerprint de l'image. C'est un SHA-1 checksum en base64 encoding.*

Search Criteria

Appliance Summary



The screenshot shows the 'Metadata' page of the StratusLab Marketplace. The page includes a navigation bar with links for Home, Endorsers, Query, Upload, and About. The main content area features a search interface with filters for OS, version, architecture, and endorser, along with a 'Sort by' dropdown and a 'Show 10 entries' option. Three appliance listings are visible, each with a download icon:

- ttylinux v14.0 x86\_64**  
Endorser: airaj@lal.in2p3.fr  
Identifier: BN1EEkPIBx87\_uLj2-sdybSI-Xb  
Created: 2012-10-22T10:34:15Z  
Description: ttylinux v14.0 x86\_64 base image. Uses the standard StratusLab contextualization mechanisms.  
[More...](#)
- openSUSE v12.1 x86\_64**  
Endorser: hudson.builder@stratuslab.eu  
Identifier: JYhKZnxPHcuGybaTTtHjcRachS  
Created: 2012-10-22T10:31:35Z  
Description: openSUSE 12.1 base image automatically created by hudson. Configured only with a root user. The firewall in the image is disabled, IPV6 is enabled, and SELinux disabled. Uses the standard StratusLab contextualization mechanisms. A swap volume is expected to be provided on /dev/sdb.  
[More...](#)
- BIO compute node**  
Endorser: christophe.blanchet@ibcp.fr  
Identifier: P4RS7s\_ZBAI\_m440OHsgk0mu4nD  
Created: 2012-09-26T09:22:37Z

# StratusLab : Details d'une Appliance

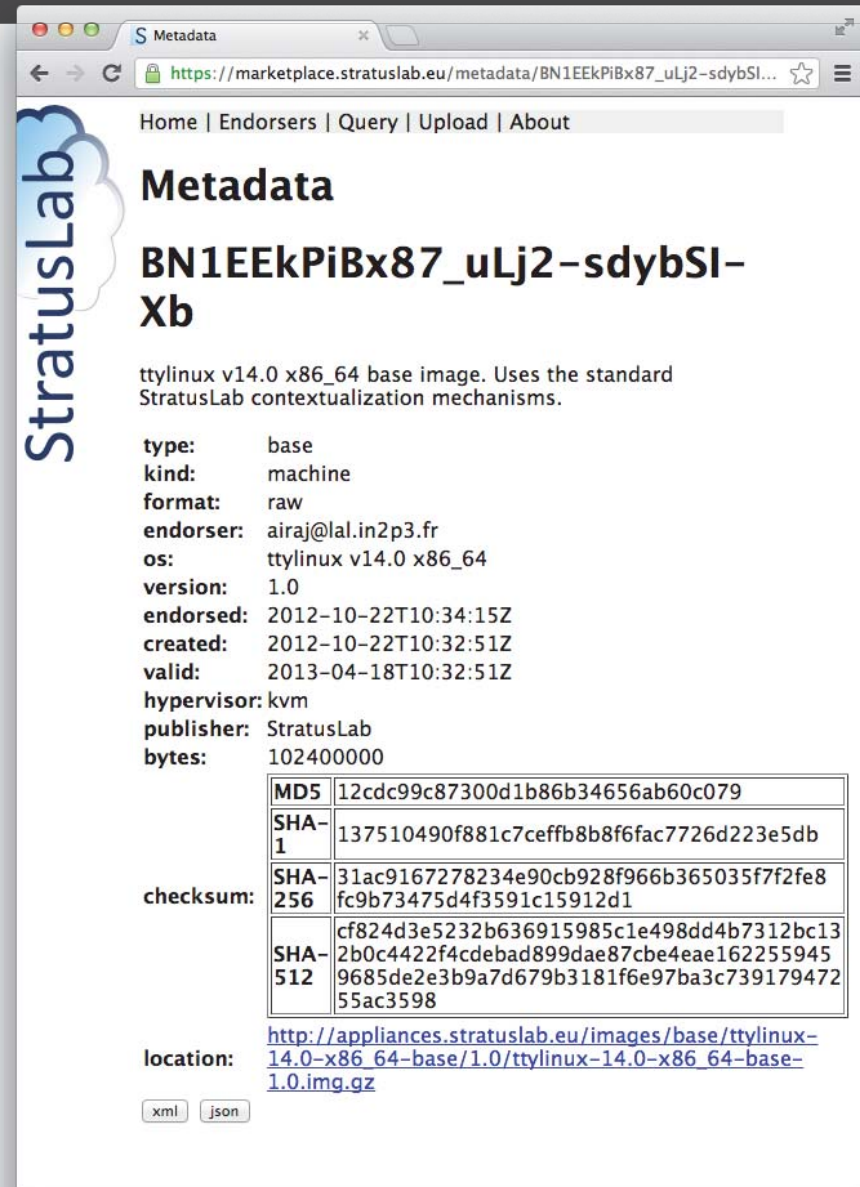
Identifier

Description

Detailed Info.

Location(s)

Other formats



The screenshot shows a web browser window displaying the metadata for a StratusLab appliance. The URL is [https://marketplace.stratuslab.eu/metadata/BN1EEkPiBx87\\_uLj2-sdybSI...](https://marketplace.stratuslab.eu/metadata/BN1EEkPiBx87_uLj2-sdybSI...). The page title is "Metadata" and the specific appliance ID is "BN1EEkPiBx87\_uLj2-sdybSI-Xb". The description states it is a "ttylinux v14.0 x86\_64 base image" that uses standard StratusLab contextualization mechanisms. The metadata includes fields for type, kind, format, endorser, os, version, endorsed, created, valid, hypervisor, publisher, and bytes. A checksum section contains a table with MD5, SHA-1, SHA-256, and SHA-512 hashes. The location is provided as a URL to the image file, and there are buttons for downloading in XML and JSON formats.

Home | Endorsers | Query | Upload | About

## Metadata

### BN1EEkPiBx87\_uLj2-sdybSI-Xb

ttylinux v14.0 x86\_64 base image. Uses the standard StratusLab contextualization mechanisms.

**type:** base  
**kind:** machine  
**format:** raw  
**endorser:** airaj@lal.in2p3.fr  
**os:** ttylinux v14.0 x86\_64  
**version:** 1.0  
**endorsed:** 2012-10-22T10:34:15Z  
**created:** 2012-10-22T10:32:51Z  
**valid:** 2013-04-18T10:32:51Z  
**hypervisor:** kvm  
**publisher:** StratusLab  
**bytes:** 102400000

<b>MD5</b>	12cdc99c87300d1b86b34656ab60c079
<b>SHA-1</b>	137510490f881c7ceffb8b8f6fac7726d223e5db
<b>SHA-256</b>	31ac9167278234e90cb928f966b365035f7f2fe8fc9b73475d4f3591c15912d1
<b>SHA-512</b>	cf824d3e5232b636915985c1e498dd4b7312bc132b0c4422f4cdebad899dae87cbe4eae1622559459685de2e3b9a7d679b3181f6e97ba3c73917947255ac3598

**checksum:**

**location:** [http://appliances.stratuslab.eu/images/base/ttylinux-14.0-x86\\_64-base/1.0/ttylinux-14.0-x86\\_64-base-1.0.img.gz](http://appliances.stratuslab.eu/images/base/ttylinux-14.0-x86_64-base/1.0/ttylinux-14.0-x86_64-base-1.0.img.gz)

# Transport d'Appliances

- Dans un environnement de cloud computing fédéré, les appliances sont généralement sauvegardées en dehors d'une infrastructure de cloud particulière.
  - Donc, doivent être transportées vers le cloud avant instantiation.
- Pour la plupart des distributions de cloud computing, l'utilisateur doit le faire manuellement.
- Heureusement, Certains outils existent pour faciliter un transfert automatisé.



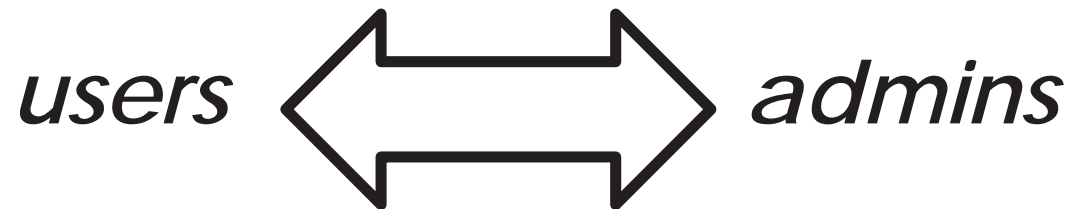
# Transport d'Appliances : Exemple d'HEPiX

- L'outil vmcaster/vmcatcher développé au sein du Groupe de travail de virtualisation HEPiX, utilise le concept de souscription à une liste d'appliances.
- Il permet le téléchargement et le transport d'une appliance de la liste comme le fait un système de gestionnaire de paquets.
- Les appliances téléchargées sont vérifiées (signatures X509) et mises en cache.

# Transport d'Appliances : Exemple de StratusLab

- En StratusLab, les appliances sont transportées à partir d'un serveur web ou de stockage dans le Cloud.
- Sur la base de l'identifiant de l'appliance dans le marketplace, le transport de l'appliance est effectué de manière transparente par l'infrastructure de Cloud.
- Les appliances téléchargées sont ensuite vérifiées et mises en cache dans le disque persistant, de façon à ce que le transport d'une appliance ne se fait qu'une seule fois.

# Autorisation d'appliances



- Démarrez l'instanciation d'une VM via son identifiant. Dans StratusLab, par exemple :

- `stratus-run-instance`  
`LwCRbwCa1YSysY1wftQdA`  
`j6Bwoi`

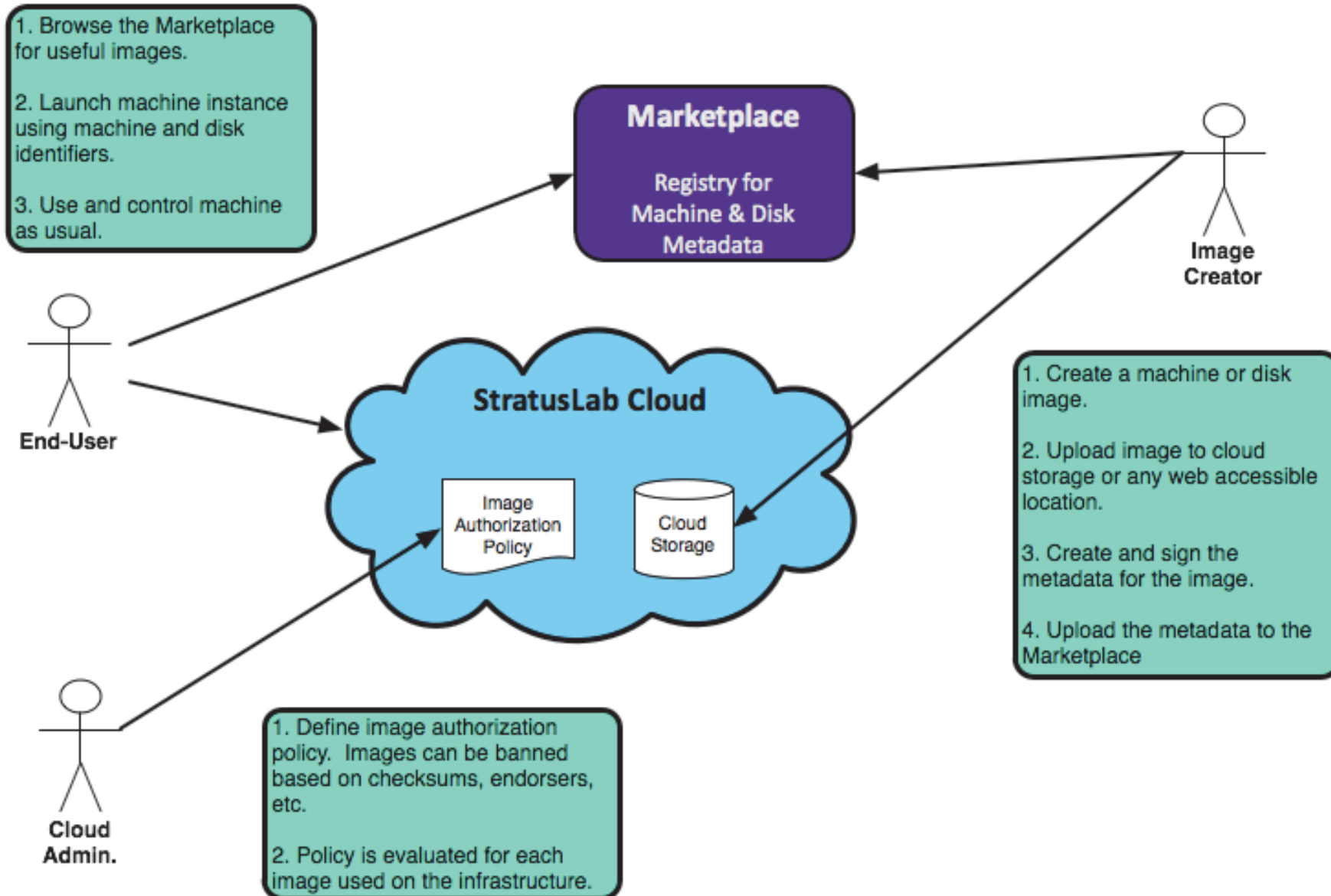
- StratusLab valide l'image avant son déploiement :
  - `stratus-policy-image` applique une politique définie par l'administrateur
  - Politique peut inclure l'endorser white lists, checksum black lists, etc.

# Publier, partager des appliances : différents acteurs.

## ▣ Avantages

- ▣ **Utilisateurs** : chercher et utiliser des images existantes
- ▣ **Créateurs** : publier leurs images et attirer plus d'utilisateurs
- ▣ **Administrateurs** : évaluer la fiabilité des images avec les métadonnées

# Acteurs et Workflows : Exemple de StratusLab



# Conclusions

- Les appliances se différencient par leur formats d'images et leurs mécanismes de contextualisation
- Le cycle de vie et gestion des appliances contient les étapes suivantes :
  - Création des appliances
  - Stockage des appliances
  - Transport des appliances
  - Enregistrement des appliances (métadonnées)
  - Politique des autorisations des appliances
- Le fédération de Cloud a mené naturellement vers une discussion sur un compromis de :
  - Standardisation des formats des appliances et / ou prise en charge de multiples formats par les infrastructure de Cloud
  - Standardisation d'outils de contextualisation

# Questions et Discussion