MBB - Workflow de réservation de nos machines SMP

Rémy Dernat

12 décembre 2017

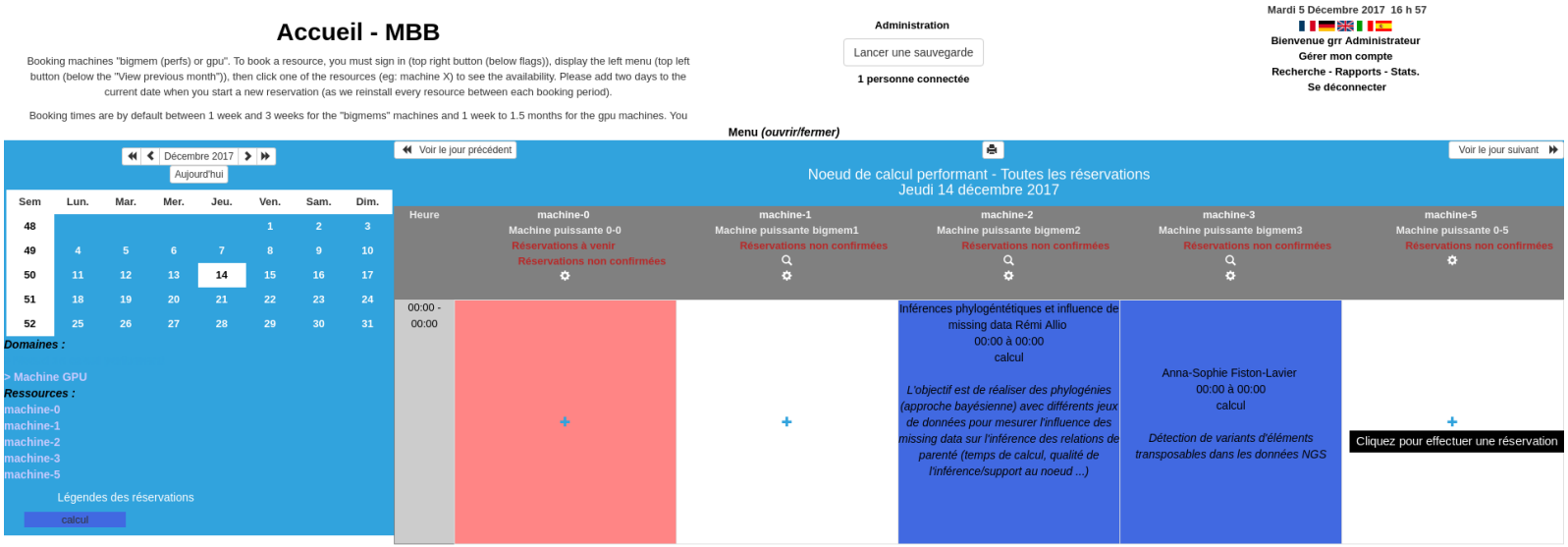
Machines réservables

- 3 **bigmems** / ~**Dell PE R815** : *512Go RAM / 64 coeurs et qqs To de stockage en local*
- 2 machines GPU type DevBox (3 à 4 cartes TitanXp)

Fonctionnement des réservations des machines performantes SMP - "bigmems"

1. Réservation des utilisateurs dans GRR,
2. Création d'un évènement dans Caldav (NextCloud) avec en résumé l'utilisateur et la date de commencement après validation par un admin de l'évènement dans GRR,
3. Cron qui s'exécute sur un serveur SaltStack toutes les nuits pour vérifier si une réservation ne commence pas le jour même,
4. Si une réservation commence (après analyse du résumé de l'évènement caldav), alors l'orchestrateur SaltStack va demander la réinstallation de la machine.

GRR


Accueil - MBB
Booking machines "bigmem (perfs) or gpu". To book a resource, you must sign in (top right button (below flags)), display the left menu (top left button (below the "View previous month")), then click one of the resources (eg: machine X) to see the availability. Please add two days to the current date when you start a new reservation (as we reinstall every resource between each booking period).
Booking times are by default between 1 week and 3 weeks for the "bigmems" machines and 1 week to 1.5 months for the gpu machines. You
Administration
Lancer une sauvegarde
1 personne connectée
Mardi 5 Décembre 2017 16 h 57
Bienvenue grr Administrateur
Gérer mon compte
Recherche - Rapports - Stats.
Se déconnecter
Menu (ouvrir/fermer)
Décembre 2017
Aujourd'hui
Sem Lun. Mar. Mer. Jeu. Ven. Sam. Dim.
48 1 2 3
49 4 5 6 7 8 9 10
50 11 12 13 14 15 16 17
51 18 19 20 21 22 23 24
52 25 26 27 28 29 30 31
Domaines :
> Machine GPU
Ressources :
machine-0
machine-1
machine-2
machine-3
machine-5
Légendes des réservations
calcul
Voir le jour précédent
Voir le jour suivant
Noeud de calcul performant - Toutes les réservations
Jeudi 14 décembre 2017
Heure
machine-0
Machine puissante 0-0
Réservations à venir
Réservations non confirmées
machine-1
Machine puissante bigmem1
Réservations non confirmées
machine-2
Machine puissante bigmem2
Réservations non confirmées
machine-3
Machine puissante bigmem3
Réservations non confirmées
machine-5
Machine puissante 0-5
Réservations non confirmées
00:00 - 00:00
Inférences phylogéntétiques et influence de missing data Rémi Allio
00:00 à 00:00
calcul
L'objectif est de réaliser des phylogénies (approche bayésienne) avec différents jeux de données pour mesurer l'influence des missing data sur l'inférence des relations de parenté (temps de calcul, qualité de l'inférence/support au noeud ...)
Anna-Sophie Fiston-Lavier
00:00 à 00:00
calcul
Détection de variants d'éléments transposables dans les données NGS
Cliquez pour effectuer une réservation

caldav - nextcloud

Fonctionnement des réservations des machines performantes SMP - "bigmems"

- L'orchestrateur SaltStack
  1. `https://docs.saltstack.com/en/latest/topics/orchestrate/orchestrate_runner.html#orchestrate-runner`
  2. Met à jour les données du serveur Salt et les serveurs TFTP (FAI `https://fai-project.org/`) pour ce client,
  3. Arrête les services clés sur le client, démonte les filesystem,
  4. Ordonne à la machine de rebooter,
  5. Supprime la clé du client à réinstaller,
  6. Attend (longtemps - après FAI) que la machine se réinstalle,

Orchestrateur SaltStack - extrait

root@newthaler: ~/un-peu-de-sel/salt_states/orch

```
bigmem_init.sls  modify_container_values.sls  node_init.sls
root@newthaler:~/un-peu-de-sel/salt_states/orch# cat bigmem_init.sls
{% set host = salt.pillar.get('reinstall') %}
{% set master = 'newthaler' %}
{% set master_cluster = 'cluster-mbb.mbb.univ-montp2.fr' %}

{% set container_str = 'bigmems:' ~ host ~ ':container:name' %}
{% set the_container = salt.pillar.get(container_str) %}

set_bigmem_to_install:
  salt.state:
    - tgt: 'faiserv'
    - sls:
      - tftp

stop_lxc_if_running:
  salt.function:
    - name: cmd.run
    - tgt: {{ host }}
# stopping lxd stop all containers
    - arg:
      - lxd shutdown

# force to unmount all nfs storage locally before rebooting
stop_nfs_mounts:
  salt.state:
    - tgt: {{ host }}
    - sls:
      - mount.umount_f

reboot:
  salt.function:
    - name: system.reboot
    - tgt: {{ host }}
```

FAI - Class

root@faiserv: /srv/fai/config/class

```
root@faiserv:/srv/fai/config/class# ls
10-base-classes  40-parse-profiles.sh  60-misc     COMPUTE.var  example.profile  FRENCH.var  INSTALL.var    ISEMNODE.var  SYSINFO.var
20-hwdetect.sh   50-host-classes       CENTOS.var  DEBIAN.var   FAIBASE.var      GERMAN.var  INVENTORY.var  MBB.var
root@faiserv:/srv/fai/config/class# cat 50-host-classes
#! /bin/bash

# assign classes to hosts based on their hostname

# do not use this if a menu will be presented
[ "$flag_menu" ] && exit 0

# use a list of classes for our demo machine
case $HOSTNAME in
    faiserver)
        echo "FAIBASE DEBIAN DEMO FAISERVER" ;;
    demohost|client*)
        echo "FAIBASE DEBIAN DEMO" ;;
    xfcehost)
        echo "FAIBASE DEBIAN DEMO XORG XFCE LVM";;
    gnomehost)
        echo "FAIBASE DEBIAN DEMO XORG GNOME";;
    centos)
        echo "FAIBASE CENTOS"   # you may want to add class XORG here
        ifclass I386 && echo CENTOS6_32  # AFAIK there's no 32bit C7
        ifclass AMD64 && echo CENTOS7_64
        exit 0 ;; # CentOS does not use the GRUB class
    ubu*)
        echo "FAIBASE UBUNTU XENIAL64 DESKTOP MBB FRENCH SALTSTACK"
        exit 0 ;;
    compute*)
        echo "FAIBASE UBUNTU XENIAL64 COMPUTE FRENCH LXD SALTSTACK BIOTOOLS"
        exit 0 ;;
    isemnode*)
        echo "FAIBASE UBUNTU XENIAL64 ISEMNODE FRENCH LXD SALTSTACK BIOTOOLS"
        exit 0 ;;
    trust*)
        echo "FAIBASE UBUNTU TRUSTY64 DESKTOP MBB FRENCH XFCE SALTSTACK"
        exit 0 ;;
```

Fonctionnement des réservations des machines performantes SMP - "bigmems"

- L'orchestrateur SaltStack
  1. Re-rajoute la machine dans Salt (rajout de la clé),
  2. Reformate les autres disques et les monte en Zfs,
  3. Déploie toutes les recettes particulières pour la machine,
- Le lendemain matin : envoie d'un mail à l'utilisateur pour vérifier que tout est Ok (son login, ses accès, ses points de montage, ...).
  Actuellement cette sous-partie n'est pas totalement automatisée afin de vérifier manuellement que tout s'est bien installé.

Discussion

- Temps nécessaire pour développer ce workflow => Beaucoup de temps (...), car :
  - Légère modification de GRR pour créer des évènements caldav dans NextCloud,
  - Création d'un client caldav en bash (...),
  - Installation et configuration de FAI,
  - + beaucoup de config Salt pour gérer orchestrateur, recettes DHCP + TFTP pour FAI + recettes spécifiques de la machine.

Discussion

- J'aurai pu limiter le travail de Salt avec plus de configurations FAI,
  Mais : étapes de post-install préférables avec Salt car plus homogène avec le reste du parc (pour tout avoir dans Salt),
- L'insertion dans CalDav permet de voir les réservations sans aller dans GRR, mais en utilisant n'importe quel client caldav (ex : lightning/thunderbird),
- Réinstallation complète >1h00,
- Connexion cluster - noeud de calcul type conteneurs LXD sur bigmems,
- Environnement "propre" entre chaque utilisation,

Discussion

- Nouvelles machines GPU réservables - Utilisation de conteneurs Docker (*nvidia-docker*) et développement de quelques services spécifiques ; beaucoup plus simple que tout ce qui a été fait précédemment, mais pas du bare-metal.
- Réutilisation des recettes précédentes + *Banquise* pour gérer nos futurs clusters de calcul.

Contacts

- remy.dernat@umontpellier.fr