

## Hobo - Development #9821

### Accélérer le traitement des notifications

01 février 2016 11:32 - Benjamin Dauvergne

<b>Statut:</b>	Fermé	<b>Début:</b>	01 février 2016
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Benjamin Dauvergne	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	
<b>Patch proposed:</b>	Oui		
<b>Description</b>			
Actuellement hobo-agent lance un process Django pour chaque notification reçu, le temps de traitement et en moyenne d'1,5s dont 90% doit être de l'initialisation (python, Django, etc..).			
Il faudrait réfléchir à rendre les processus traitants pérennes pour ne plus supporter ce temps d'initialisation sur chaque message.			
Une première idée serait de créer une nouvelle commande Django 'batch-execute' qui accepterait sur l'entrée standard des commandes ainsi que des flux d'entrée et exécuterait ces commandes sans jamais terminer le processus.			
<b>Demandes liées:</b>			
Lié à w.c.s. - Development #9822: Avoir une commande hobo_batch qui accepte d...		<b>Rejeté</b>	<b>01 février 2016</b>
Lié à Hobo - Development #9932: Cache paresseux de package_version		<b>Fermé</b>	<b>10 février 2016</b>

#### Révisions associées

##### Révision d1780b0a - 16 mars 2016 14:14 - Frédéric Péters

worker: use actual tenants to determine if hobo\_notify is relevant (#9821)

#### Historique

##### #1 - 01 février 2016 14:22 - Benjamin Dauvergne

- Lié à Development #9822: Avoir une commande hobo\_batch qui accepte des commandes venant d'un pipe ajouté

##### #2 - 01 février 2016 15:00 - Benjamin Dauvergne

- Fichier 0001-add-new-command-hobo\_batch-got-agent.common-to-launc.patch ajouté

- Fichier 0002-use-hobo\_natch-in-agent-Service-classes-fixes-9821.patch ajouté

- Patch proposed changé de Non à Oui

Le protocole avec la commande hobo\_batch est assez simple:

```
WRITE len(args)::uint32-little-endian
POUR CHAQUE ARG:
  WRITE len(arg)::uint32-little-endian
  WRITE arg
WRITE len(input)::uint32-little-endian
WRITE input
exit_code <- READ uint32-little-endian
len_stdout <- READ uint32-little-endian
stdout <- READ len_stdout
```

##### #3 - 01 février 2016 15:00 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

C'est pas byzance mais ça va un peu plus vite:

```
[2016-02-01 14:59:51,278: INFO/MainProcess] Received task: hobo-notify[68f34ebc-738a-420d-9695-9a9639e63ca5] e
xpires:[2016-02-01 14:01:50.830151+00:00]
[2016-02-01 14:59:51,313: INFO/MainProcess] Received task: hobo-notify[c9d0aade-9832-4239-a5de-9f911acd2d87] e
xpires:[2016-02-01 14:01:51.302349+00:00]
[2016-02-01 14:59:51,331: INFO/MainProcess] Received task: hobo-notify[b9c4ed8e-393e-4a85-acd8-8e2cbf056769] e
```

```
xpires:[2016-02-01 14:01:51.319604+00:00]
[2016-02-01 14:59:51,349: INFO/MainProcess] Received task: hobo-notify[cc5b2fbc-97d2-4bcc-9aa1-f41ccd355d19] e
xpires:[2016-02-01 14:01:51.337943+00:00]
[2016-02-01 14:59:51,353: INFO/MainProcess] Task hobo-notify[68f34ebc-738a-420d-9695-9a9639e63ca5] succeeded i
n 0.0739311221987s: None
[2016-02-01 14:59:51,354: INFO/MainProcess] Task hobo-notify[c9d0aade-9832-4239-a5de-9f911acd2d87] succeeded i
n 0.000980818644166s: None
[2016-02-01 14:59:51,371: INFO/MainProcess] Received task: hobo-notify[f09971ab-06d6-46da-9ddf-6c381b343137] e
xpires:[2016-02-01 14:01:51.359202+00:00]
[2016-02-01 14:59:51,392: INFO/MainProcess] Received task: hobo-notify[2ae69c65-6d4b-4dbf-afe5-8ce1188fe367] e
xpires:[2016-02-01 14:01:51.381798+00:00]
[2016-02-01 14:59:51,604: INFO/MainProcess] Task hobo-notify[b9c4ed8e-393e-4a85-acd8-8e2cbf056769] succeeded i
n 0.248861344531s: None
[2016-02-01 14:59:51,736: INFO/MainProcess] Task hobo-notify[cc5b2fbc-97d2-4bcc-9aa1-f41ccd355d19] succeeded i
n 0.131633743644s: None
[2016-02-01 14:59:51,808: INFO/MainProcess] Task hobo-notify[f09971ab-06d6-46da-9ddf-6c381b343137] succeeded i
n 0.0714618433267s: None
[2016-02-01 14:59:51,970: INFO/MainProcess] Task hobo-notify[2ae69c65-6d4b-4dbf-afe5-8ce1188fe367] succeeded i
n 0.161524331197s: None
```

#### #4 - 01 février 2016 15:01 - Benjamin Dauvergne

Ces patchs ainsi que celui pour w.c.s. sont déployés sur cam-dev.

#### #5 - 01 février 2016 18:15 - Frédéric Péters

Il y a des chiffres sur la situation dont on vient ?

Ça m'ennuie vraiment vraiment de partir sur quelque chose qui va bien complexifier l'affaire (plus de code, process à conserver, debug plus difficile, etc.)

Dans [#9822](#), il est écrit :

À Alfortville le parcours welco -> authentic -> wcs -> welco prends beaucoup trop de temps (30s) lors de la création d'un nouveau contact, ce développement devrait permettre d'en gagner sur la réception des notifications.

Sur le parcours Welco → Authentic → w.c.s. → Welco, la partie hobo c'est juste le lien "Authentic → w.c.s.", si la chaîne totale prend 30 secondes et que le traitement d'un hobo-notify prend 1,5 secondes, on est sûr de viser le bon endroit avec ce changement proposé ?

(j'ai l'impression qu'on s'emballe sur un sujet méritant plus de temps et d'analyse).

#### #6 - 01 février 2016 18:35 - Benjamin Dauvergne

La moyenne sur montpellier-dev est de 4.2s par notification, là on est au pire 20 fois plus rapides.

#### #7 - 01 février 2016 19:15 - Frédéric Péters

Optimiser 4 secondes sur un lot qui en fait 30, ça reste pour moi viser le mauvais endroit.

Mais là on se met à parler de 4 secondes alors qu'il est écrit dans la description du ticket 1,5 secondes.

Et puis on parle maintenant de la dev de Montpellier, où toutes les applications sont notifiées/démarrées en série alors que la cible Alfortville est différente, la notification qui arrive sur le serveur wcs lance un seul processus. (et le hobo-notify y est noté comme prenant ~2 secondes). (je propose d'aller discuter de ça sur le ticket exposant le problème complet initial avant de partir sur des modifications techniques). (problème qui est exposé plusieurs heures après l'arrivée de patchs techniques, fascinante chronologie) (#9825).

#### #8 - 10 février 2016 09:09 - Frédéric Péters

- Fichier 0001-worker-use-actual-tenants-to-determine-if-hobo\_notif.patch ajouté

À reprendre un peu, avec le temps d'analyse, qui reste à prolonger.

1. pour lancer le traitement d'un hobo\_notify, il y a démarrage d'un processus et le chargement de python et des différents modules prend du temps.

Dans ce ticket, idée d'avoir un processus résidant supplémentaire. Mais peut-être pourrait-on se dire que l'application déjà chargée, elle existe déjà pour l'http, un middleware hobo pourrait ajouter le traitement d'un POST sur un /hobo\_notify (qu'on pourrait limiter à localhost), peut-être; plutôt qu'un nouveau processus et protocole.

L'autre piste sur le sujet c'est améliorer la vitesse de démarrage des commandes. C'est ce qui a été fait côté w.c.s. Côté applications django il y a mise en cache au chargement de VersionMiddleware.get\_packages\_version(), ce qui prend vraiment du temps (chez moi, 1 seconde, sur un temps total de 1,5 seconde...).

2. il y a trop de hobo\_notify, traités par tout le monde.

Il y a [#9644](#) pour agréger les hobo\_notify.

Aussi, settings.AGENT\_HOST\_PATTERNS est utilisé pour ignorer des hobo\_notify dans l'agent même, c'est bien mais en fait AGENT\_HOST\_PATTERNS a été créé pour le hobo\_deploy où il est important de noter qu'il se combine avec le type de l'application. (sur le site combo, même s'il est mis "\*.entrouvert.org", il ne va pas traiter le déploiement d'un fargo).

Aujourd'hui il y a un hobo\_notify par service (ok, parce que les rôles peuvent être différents entre rôles), mais les notifications pour tous les sites sont reçus sur tous les serveurs, et le serveur combo qui voit arriver une requête pour "fargo-alfortville.entrouvert.org", ça va matcher "\*.entrouvert.org" et un hobo\_notify va être lancé. Sur un déploiement avec cinq services, ça fait donc 80% de notifications inutiles.

Comme piste pour ce point, hobo pourrait découvrir par lui-même les tenants existants, en déterminant le TENANT\_BASE des différentes applications (au démarrage du worker) et en faisant un os.listdir() lors de la réception des notifications. (patch proof of concept attaché)

#### #9 - 10 février 2016 09:57 - Benjamin Dauvergne

- Lié à Development #9932: Cache paresseux de package\_version ajouté

#### #10 - 10 février 2016 15:56 - Frédéric Péters

Aussi, il y a un problème de saturation amené par la synchro LDAP qui fait un save() sur tous les utilisateurs (même quand il n'y a pas eu d'attribut changé) ce qui amène un hobo\_notify. (dans [#9644](#) il est noté qu'ils pourraient être agrégés mais quand il n'y a pas de modif on pourrait ne pas faire de .save()).

#### #11 - 13 mars 2016 11:17 - Benjamin Dauvergne

Le problème de la synchro LDAP est réglé.

#### #12 - 16 mars 2016 14:14 - Frédéric Péters

- Fichier 0001-worker-use-actual-tenants-to-determine-if-hobo\_notif.patch ajouté

- Statut changé de Nouveau à En cours

#### #13 - 16 mars 2016 14:15 - Frédéric Péters

Plutôt qu'une commande get\_tenants\_dir, simplement avoir les répertoires dans les settings.

#### #14 - 16 mars 2016 15:33 - Benjamin Dauvergne

Ack.

#### #15 - 16 mars 2016 15:35 - Frédéric Péters

- Statut changé de En cours à Résolu (à déployer)

```
commit d1780b0a87a3af4805fdca75f37b936db697ef51
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Wed Feb 10 09:08:32 2016 +0100
```

```
worker: use actual tenants to determine if hobo_notify is relevant (#9821)
```

#### #16 - 23 décembre 2018 16:05 - Frédéric Péters

- Statut changé de Résolu (à déployer) à Solution déployée

### Fichiers

0001-add-new-command-hobo_batch-got-agent.common-to-launc.patc	2,88 ko	01 février 2016	Benjamin Dauvergne
0002-use-hobo_natch-in-agent-Service-classes-fixes-9821.patch	5,11 ko	01 février 2016	Benjamin Dauvergne
0001-worker-use-actual-tenants-to-determine-if-hobo_notif.patch	3,86 ko	10 février 2016	Frédéric Péters
0001-worker-use-actual-tenants-to-determine-if-hobo_notif.patch	5,3 ko	16 mars 2016	Frédéric Péters