

Combo - Development #62678

cache trop long sur la cellule "Demandes à traiter" (WcsCareFormsCell)

11 mars 2022 10:22 - Benjamin Dauvergne

Statut:	Fermé	Début:	11 mars 2022
Priorité:	Normal	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Cf. #62566			
Le cache est à 120 secondes, pour des demandes rapidement traitées (validation de ceci ou cela), c'est beaucoup trop long. Il faudrait rendre le cache configurable au niveau de la cellule ou le baisser complètement (je n'ai pas d'idée sur le bon seuil), ou trouver une méthode pour forcer l'invalidation de celui-ci.			
Demandes liées:			
Lié à Combo - Development #63354: wcs : exposer le paramètre cache_duration		Fermé	30 mars 2022

Historique

#2 - 11 mars 2022 11:53 - Benjamin Dauvergne

- Description mis à jour

#3 - 12 mars 2022 13:39 - Frédéric Péters

Il n'y a pas de parcours décrit, ni ici ni dans le ticket lié, je vais imaginer qu'il y a un retour contrôlé vers la page contenant dans la cellule. Il pourrait alors y avoir un ?refresh ajouté à l'URL, et en présence de ce paramètre le cache serait ignoré/vidé; on peut imaginer ça de manière globale mais aussi une variante ?refresh=slug pour concerner uniquement la cellule avec le slug donné.

Ça permet de conserver une utilité au cache.

#4 - 12 mars 2022 17:49 - Mikaël Ates

Dans mon cas d'usage la cellule est affichée sur une page qui dans le menu principal de combo. Il n'y a pas de lien vers cette page autre que celui-là.

#5 - 12 mars 2022 19:26 - Frédéric Péters

On peut donc imaginer que ton entrée de menu contienne ?refresh=le-slug, pour forcer de manière systématique le rafraichissement de la cellule en question. Ça marche pour ton cas, ça garde le cache effectif pour le cas commun qui n'est pas un agent faisant des allers-retours rapides vers sa page d'accueil et ça ne charge pas non plus l'interface de configuration de ces cellules (ce qui est à éviter en attendant un dispositif efficace permettant aux options "rares/avancées" de ne pas prendre de place).

#6 - 12 mars 2022 21:25 - Frédéric Péters

On peut donc imaginer que ton entrée de menu contienne ?refresh=le-slug,

Ah non je pensais menu barre latérale portail agent parce que cette cellule reste prévue pour cet usage mais ici ça n'est pas le cas on est sur une exploitation en front de la cellule.

#7 - 23 mars 2022 11:01 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#8 - 29 mars 2022 18:24 - Benjamin Dauvergne

- Fichier 0001-misc-allow-refreshing-request-s-cache-for-an-user-62.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

J'introduis un nouveau cache qui utilise le cache Django classique mais génère des clés versionnés sur plusieurs niveau. J'utilise ce nouveau cache

dans requests_wrapper.Requests si la requête est dépendante de l'utilisateur.

Dans le cas du cache d'une requête GET, la clé sera f'{md5(url)}:user:{user.pk}:' + str(cache.get(f'user:{user.pk}') or 0) ou md5(url) comme avant si l'URL ne dépend pas de l'utilisateur. Ensuite il y a l'ajout d'un nouveau middleware 'invalidate_cache_middleware' qui accepte un paramètre '?invalidate-user-cache' sur n'importe quelle URL, dans ce cas la clé 'user-{request.user.pk}' sera incrémentée (par défaut elle vaudra 0), invalidant toutes les clés liées.

#9 - 29 mars 2022 18:37 - Benjamin Dauvergne

branche à jour avec correction à des tests et pour pylint.

#10 - 29 mars 2022 19:14 - Frédéric Péters

Et dans ton idée comment somehow cette url ?invalidate-user-cache se trouve appelée ? C'est ajouté dans le workflow ?

#11 - 29 mars 2022 21:07 - Frédéric Péters

Et dans ton idée comment somehow cette url ?invalidate-user-cache se trouve appelée ? C'est ajouté dans le workflow ?

Non ça ne peut pas être ça c'est basé sur l'utilisateur de la requête request.user.pk; hésitation peut-être via PublikAuthentication mais ça dépendrait de DRF or ici il n'y a pas ça.

#12 - 29 mars 2022 21:52 - Benjamin Dauvergne

J'étais toujours sur l'idée qu'on revenait à un moment de w.c.s. vers combo et donc qu'on pouvait passer ce paramètre à ce moment (quelque soit la page retour) PS: que ce soit un lien dans la page (menu) ou une action de redirection dans le workflow, l'action de redirection dans le workflow me semblant le moyen le plus sûr de garantir le parcours et le passage du paramètre effectivement.

#13 - 29 mars 2022 22:23 - Frédéric Péters

De ce que je comprenais au moment de #62678#note-6 il n'y avait justement pas de parcours fléché, qu'il y avait clic dans la navigation principale du site, que justement ça rendait ce jeu de paramètre supplémentaire impraticable. (cf première vidéo de <https://portail-mates.test.entrouvert.org/> où les parcours se font pas mal en cliquant sur "espace pro" dans la navigation principale).

#14 - 29 mars 2022 23:39 - Benjamin Dauvergne

- Fichier 0002-public-add-a-one-pixel-gif-view-for-calling-refresh-.patch ajouté

- Fichier 0001-misc-allow-refreshing-request-s-cache-for-an-user-62.patch ajouté

Voilà avec une vue renvoyant une image gif d'un pixel transparent appellable de n'importe où. Good ?

```

```

#15 - 30 mars 2022 09:04 - Frédéric Péters

Il faudrait expliciter davantage l'idée, tu écris "appellable de n'importe où" mais tu imagines que ça soit utilisé où ? ça serait mis dans une action d'ajout d'un message dans l'historique ?

Je suis vraiment sur l'impression que se construit ici quelque chose dont l'exploitation va être trop compliquée.

(pour moi il y aurait à profiter de l'organisation à venir du paramétrage des cellules pour juste y taper un onglet cache avec une option pour le configurer, permettre des "options plus rares mais utiles dans certains contextes" c'était totalement l'idée #62965, ça matche ici).

#16 - 30 mars 2022 12:04 - Benjamin Dauvergne

Frédéric Péters a écrit :

Il faudrait expliciter davantage l'idée, tu écris "appellable de n'importe où" mais tu imagines que ça soit utilisé où ? ça serait mis dans une action d'ajout d'un message dans l'historique ?

Oui idéalement, ça évite de se demander si la personne va suivre un lien ici où là.

Je suis vraiment sur l'impression que se construit ici quelque chose dont l'exploitation va être trop compliquée.

Ça consiste à ajouter un tag img.

(pour moi il y aurait à profiter de l'organisation à venir du paramétrage des cellules pour juste y taper un onglet cache avec une option pour le configurer, permettre des "options plus rares mais utiles dans certains contextes" c'était totalement l'idée #62965, ça matche ici).

Ça revient à supprimer le cache dans certaines situations, je ne sais pas si c'est mieux mais ça me va d'ouvrir un ticket pour exposer `cache_duration` (je ne vois pas bien le lien avec [#62965](#) à part que oui la présentation changera), ça faisait partie de mes propositions.

#17 - 30 mars 2022 12:16 - Benjamin Dauvergne

- Lié à [Development #63354](#): `wcs` : exposer le paramètre `cache_duration` ajouté

#18 - 30 mars 2022 12:17 - Benjamin Dauvergne

Après je trouve que l'invalidation du cache sur les appels HTTP est un problème récurrent de toute façon (sur les cellules JSON aussi) et que le résoudre par "on supprime le cache" ce n'est pas vraiment le résoudre AMHO. Là on avait un début de solution un peu global.

#19 - 30 mars 2022 12:50 - Frédéric Péters

Oui idéalement, ça évite de se demander si la personne va suivre un lien ici où là.
Ça consiste à ajouter un tag `img`.

Perso je lis ça comme explication à donner : si vous voulez invalider le cache ajoutez une action "message dans l'historique" dans laquelle vous aurez à copié/collé `` et puis sans doute ça va pourrir un peu l'affichage de l'historique avec des commentaires qui apparaîtront vide mais à chaque jour sa peine.

C'est pour moi une exploitation compliquée, ça se trouvera deux fois utilisé au fond d'un workflow puis plus jamais.

je ne vois pas bien le lien avec [#62965](#)

Il y a une certaine opposition à ajouter des options rarement utilisées et qui encombrant l'interface. [#62965](#) est le chemin pour conserver une configuration avec les options utiles, tout en se permettant des options plus rares.

je trouve que l'invalidation du cache sur les appels HTTP est un problème récurrent de toute façon

Je suis d'accord.

Là on avait un début de solution un peu global

"un peu global", pas assez, le cadre de réflexion doit être plus large. Ce patch vient en ignorant totalement les problèmes d'invalidation de cache qui peuvent exister dans `publik` famille, par exemple.

#20 - 30 mars 2022 13:03 - Benjamin Dauvergne

Frédéric Péters a écrit :

Oui idéalement, ça évite de se demander si la personne va suivre un lien ici où là.
Ça consiste à ajouter un tag `img`.

Perso je lis ça comme explication à donner : si vous voulez invalider le cache ajoutez une action "message dans l'historique" dans laquelle vous aurez à copié/collé `` et puis sans doute ça va pourrir un peu l'affichage de l'historique avec des commentaires qui apparaîtront vide mais à chaque jour sa peine.

C'est pour moi une exploitation compliquée, ça se trouvera deux fois utilisé au fond d'un workflow puis plus jamais.

Je suis d'accord que ce n'est pas super propre, mais tant que le cache est lié au front, la notification doit passer par le front, une solution via `web-service` ne pourra pas viser le bon front. Donc c'est soit ça, soit aller vers un cache cohérent entre les fronts ou trouver une donnée cohérente (genre un compteur par utilisateur stocké en base qui serait ajouté aux clés du cache, comme ici, mais le compteur lui serait global et pas local au front) qui serve de version.

Même à imaginer une solution automatique où `w.c.s.` viendrait notifier les briques de lui même que pour tel utilisateur les données ont changé (ça ne nécessiterait donc pas de prise en compte particulière par les administrateurs fonctionnels) nécessiterait de résoudre ce problème technique de fond. Ou bien on pourrait dire que dans une telle situation `w.c.s.` ajoute de lui même ce tag `` à sa page, mais ça demande plus de boulot pour identifier dans `w.c.s.` toutes les pages où cette action devra être faite (plus les cas d'action de redirection dans le workflow, où `w.c.s.` pourrait éventuellement ajouter de lui même le `?refresh-user-cache` s'il identifie une URL du portail), c'est moins de travail pour l'administrateur mais plus de complexité dans le code.

je trouve que l'invalidation du cache sur les appels HTTP est un problème récurrent de toute façon

Je suis d'accord.

Là on avait un début de solution un peu global

"un peu global", pas assez, le cadre de réflexion doit être plus large. Ce patch vient en ignorant totalement les problèmes d'invalidation de cache qui peuvent exister dans publik famille, par exemple.

Je veux bien plus de détails pour voir en quoi ça n'y répond pas pour des cellules fiches ça doit le faire tout seul aussi (pourvu qu'on mette le tag etc... que la modification vienne d'une action dans w.c.s. via un formulaire, etc..).

#21 - 30 mars 2022 13:15 - Frédéric Péters

Je veux bien plus de détails pour voir en quoi ça n'y répond pas pour des cellules fiches ça doit le faire tout seul aussi (pourvu qu'on mette le tag etc... que la modification vienne d'une action dans w.c.s. via un formulaire, etc..).

Et que tout ça soit fait par l'utilisateur.

Ça ignore les manipulations des agents, qui se trouvent à aller modifier l'info pour une famille via un formulaire et puis la modif n'apparaît pas sur la page combo associée. (résultat on y a un cache court à 5 secondes pour ignorer le problème).

#22 - 30 mars 2022 17:07 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#23 - 30 mars 2022 23:50 - Benjamin Dauvergne

- Fichier 0004-misc-add-mule-to-receive-postgres-notifications-6267.patch ajouté
- Fichier 0005-api-add-endpoint-to-invalid-user-cache-on-all-fronte.patch ajouté
- Fichier 0002-public-add-a-one-pixel-gif-view-for-calling-refresh-.patch ajouté
- Fichier 0001-misc-allow-refreshing-request-s-cache-for-an-user-62.patch ajouté
- Fichier 0003-misc-add-module-to-send-receive-postgres-notificatio.patch ajouté

Hack alert !! J'ai réussi à pondre une API qui invalide du cache sur tous les fronts, le endpoint poste une notification via Postgresql sur le topic Publik et tous les fronts gagnent une mule uwsgi qui écoute ces messages.

Donc en gros on fait, un GET sur `{{ portal_url }}/api/cache/invalidate/user/?NameID={{ session_user_nameid }}`, qui supposons correspond à User(id=1) sur combo, ça va faire la requête SQL NOTIFY publik, '{"tenant": "combo.dev.publik.love", "message": {"type": "invalidate", "key": "user:1"}}' et au final le code écoutant dans `combo.utils.notification_mule` va appeler `multilevel_cache.invalidate('user:1')` sur tous les fronts.

Je ne sais pas si c'est une bonne idée, mais ça marche (localement en lançant combo en mode uwsgi dans supervisor), on constate que `cache.get('user-1')` est bien incrémenté à chaque appel.

Rapidement testé en augmentant `cache_duration` à 3600 sur `WcsCurrentDraftsCell`, création/suppression d'un brouillon ne sont pas visible immédiatement mais le devienne après l'appel à l'API.

PS: c'est surtout pour discuter, avoir des redis synchronisés sur les fronts à la place de memcache serait plus simple.

#24 - 31 mars 2022 08:27 - Frédéric Péters

c'est surtout pour discuter

Je préférerais qu'avant de foncer dans des trucs on prenne du temps pour poser le sujet; les problèmes qu'on rencontre, ce qu'on veut atteindre, les pistes pour y arriver.

(et fermer ce ticket qui a pris une dimension qui le dépasse).

#25 - 31 mars 2022 09:11 - Benjamin Dauvergne

- Assigné à changé de Benjamin Dauvergne à Frédéric Péters

Qu'est-ce que je dis à Mike pour sa cellule ? Tout le monde a le droit à son cache de 5 seconde mais pas lui ?

#26 - 31 mars 2022 09:47 - Frédéric Péters

Il me semblait que tu avais créé [#63354](#).

Fichiers

0001-misc-allow-refreshing-request-s-cache-for-an-user-62.patch	8,36 ko	29 mars 2022	Benjamin Dauvergne
0002-public-add-a-one-pixel-gif-view-for-calling-refresh-.patch	1,42 ko	29 mars 2022	Benjamin Dauvergne
0001-misc-allow-refreshing-request-s-cache-for-an-user-62.patch	9,56 ko	29 mars 2022	Benjamin Dauvergne
0004-misc-add-mule-to-receive-postgres-notifications-6267.patch	1,48 ko	30 mars 2022	Benjamin Dauvergne
0005-api-add-endpoint-to-invalid-user-cache-on-all-fronte.patch	3,28 ko	30 mars 2022	Benjamin Dauvergne
0002-public-add-a-one-pixel-gif-view-for-calling-refresh-.patch	1,42 ko	30 mars 2022	Benjamin Dauvergne
0001-misc-allow-refreshing-request-s-cache-for-an-user-62.patch	9,56 ko	30 mars 2022	Benjamin Dauvergne
0003-misc-add-module-to-send-receive-postgres-notificatio.patch	3,65 ko	30 mars 2022	Benjamin Dauvergne