

w.c.s. - Development #60817

wscalls: pouvoir n'enregistrer que les erreurs "techniques" (et éviter "[wscalls] 200 Ok")

19 janvier 2022 13:55 - Benjamin Dauvergne

Statut:	Fermé	Début:	19 janvier 2022
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Non		
Description			
Cf. ticket remontant le dérangement dans #60814			
Les drapeaux "notify_on_errors" et "record_on_errors" mélange erreurs techniques (exceptions, code HTTP >= 400) et erreurs logiques data['err'] != 0. Et dans ce dernier cas l'erreur a un intitulé absolument pas clair de [WSCALLS] 200 Ok. Je pense qu'il serait judicieux soit de ne rien faire sur du non zéro, soit de séparer ce cas d'un autre paramétrage et d'avoir un intitulé plus clair [WSCALLS] application error "<err_desc or err>" (si err != 1).			
Demandes liées:			
Lié à w.c.s. - Development #61953: wscall: logger le nom du wscall dans l'erreur		Nouveau	18 février 2022
Lié à w.c.s. - Development #13593: wscall : notifier des erreurs applicatives...		Fermé	13 octobre 2016

Historique

#2 - 19 janvier 2022 13:56 - Benjamin Dauvergne

- *Sujet changé de wscalls: pouvoir n'enregistrer que les erreurs "techniques" à wscalls: pouvoir n'enregistrer que les erreurs "techniques" (et éviter "[wscalls] 200 Ok")*

#3 - 19 janvier 2022 14:00 - Marie Kuntz

Assez d'accord pour séparer les 2, jusqu'ici je n'avais pas compris que les "erreurs" sur code 200 correspondaient à err != 0, on en apprend tous les jours

#4 - 19 janvier 2022 14:33 - Thomas Noël

Je vote pour parvenir à enregistrer « [WSCALLS] application error "<err_desc or err>" » en cas de retour d'erreur logicielle (err != 0). Juste pour garder un peu de cohérence sur la notion d'erreur, on présente ces erreurs comme telles, alors enregistrons-les avec "notify_on_errors" et "record_on_errors"

A priori ça se joue dans wcs/wscalls.py ici:

```
if (app_error_code != 0 or status >= 400) and (notify_on_errors or record_on_errors):
    summary = '<no response>'
    if response is not None:
        summary = '%s %s' % (status, response.reason)
    get_publisher().record_error(
        summary, context='[WSCALL]', notify=notify_on_errors, record=record_on_errors
    )
```

et dans wcs/wf/wscall.py :

```
if exception is None:
    summary = '<no response>'
    if response is not None:
        summary = '%s %s' % (response.status_code, response.reason)
else:
    exc_type, exc_value = sys.exc_info()[1:]
    summary = traceback.format_exception_only(exc_type, exc_value)[-1]

get_publisher().record_error(
    error_summary=summary,
```

Dans les deux cas il faudrait ajouter l'info sur err (et err_desc + err_class s'ils sont là) en plus de juste « '%s %s' % (response.status_code, response.reason) »

#5 - 19 janvier 2022 17:08 - Benjamin Dauvergne

Du ticket lié je devine aussi un autre problème, c'est que le tech_id qui permet de regrouper les erreurs est construit à partir du résumé qui se résume malheureusement à formdef_id + '200 Ok'. Si ce webservice.xyz est utilisé dans plusieurs conditions de diverses manières, ça ne fait qu'une erreur et on a jamais le contexte du champ concerné, ni d'ailleurs le nom du webservice.

Il faudrait ajouter à record_error la possibilité de désigner le web-service, ainsi que le champ (mais le champ on devrait récupérer ça du contexte poser lors de l'évaluation de la condition d'une manière ou d'une autre).

#6 - 18 février 2022 10:31 - Benjamin Dauvergne

- Lié à *Development #61953*: wscall: logger le nom du wscall dans l'erreur ajouté

#8 - 07 octobre 2023 19:09 - Frédéric Péters

- Statut changé de *Nouveau* à *Fermé*

Je vote pour parvenir à enregistrer « [WSCALLS] application error "<err_desc or err>" » en cas de retour d'erreur logicielle (err != 0). Juste pour garder un peu de cohérence sur la notion d'erreur, on présente ces erreurs comme telles, alors enregistrons-les avec "notify_on_errors" et "record_on_errors"

C'est ce que fait [#13593](#) (presque), je viens de le rebaser et de lui ajouter dans les tests l'info précise sur le texte des erreurs logguées, qui est "[WSCALL] err: 1, err_desc: some error" mais si on veut y écrire "application error" devant, pas de soucis.

#9 - 07 octobre 2023 19:09 - Frédéric Péters

- Lié à *Development #13593*: wscall : notifier des erreurs applicatives en fournissant autre chose que le code de statut de la requête ajouté