w.c.s. - Development #4228

Actions de workflow Appel et Retour

17 janvier 2014 11:37 - Benjamin Dauvergne

Statut:	Fermé	**Début:**	17 janvier 2014
Priorité:	Normal	**Echéance:**	
Assigné à:		**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Non	**Planning:**	Non

Description

Discussion avec Thomas autour des actions globales qui a terminé sur les workflows paramétriques/procédure de workflow, l'idée:
- ajouter une pile de statuts dans les champs des formulaires
- ajouter une action Appel qui permet de sauter à un statut X tout en poussant un statut Y sur la pile (éventuellement le statut en cours)
- ajouter une action Retour qui saute au dernier statut empilé

En utilisant tout cela il devient possible de définir des sous-workflow appelables n'importe où dans le workflow principal. Au niveau visuel on représenterait cette action par une flèche vers le statut retour (le statut Y), avec comme label « Appel "X" ».

Demandes liées:

Lié à w.c.s. - Development #3659: Possibilité de définir des actions globales.	**Fermé**	**24 septembre 2013**
Lié à w.c.s. - Development #16473: Saut vers "statut précédent"	**Rejeté**	**24 mai 2017**
Lié à w.c.s. - Development #16524: Possibilité de saut "en arrière"	**Fermé**	**27 mai 2017**

Historique

#1 - 17 janvier 2014 15:56 - Benjamin Dauvergne

À la création d'un Appel on peut ajouter une validation comme quoi X est bien une procédure: tous les chemins partant de X contiennent à un moment une action Retour.
De même en posant une action Retour on peut vérifier qu'aucun parent du statut sur lequel l'action est posée n'est relié via des Jump à un statut initial.

La première condition est pour éviter qu'un appel ne revienne jamais sur Y et la deuxième qu'un Retour ait lieu alors que la pile est vide.

#2 - 25 mai 2017 15:13 - Benjamin Dauvergne

- Lié à Development #16473: Saut vers "statut précédent" ajouté

#3 - 25 mai 2017 19:21 - Frédéric Péters

- Patch proposed mis à Non

Pour garder l'idée mais l'intégrer dans les actions actuelles, plutôt que ce vocabulaire d'appel/retour/pile que je ne me vois pas documenter; on pourrait avoir dans les destinations des sauts un "Marqueur précédent" et dans l'action de saut manuel, une case à cocher "poser un marqueur sur ce statut". (?)

#4 - 25 mai 2017 19:34 - Benjamin Dauvergne

Oui ça me semble tout aussi bien, je ne pense pas que ce sera fondamentalement plus simple à intégrer conceptuellement pour un administrateur fonctionnelle mais le vocabulaire serait simplifié.

#5 - 25 mai 2017 20:40 - Frédéric Péters

- Fichier 0001-workflows-add-possibility-to-mark-and-jump-to-marked.patch ajouté

- Statut changé de Nouveau à En cours

- Patch proposed changé de Non à Oui

#6 - 26 mai 2017 00:38 - Benjamin Dauvergne

Finalement il y a un souci pour moi par rapport à l'idée call/return c'est que le call/return met en relation trois états, le statut avant l'appel, le statut après l'appel et le statut après le retour. Ici il n'y a que deux statuts celui du marqueur et celui vers lequel on saute, on va forcément revenir au marqueur et on ne saura pas si l'appel a eu lieu ou pas, pour obtenir la sémantique d'un appel on sera obligé de créer une variable par exemple et d'y stocker l'information comme quoi l'appel a déjà eu lieu et qu'il ne faut pas le refaire. Pour avoir quelque chose d'équivalent il faudrait que le marqueur

ne soit pas mis sur le statut en cours mais sur un autre, au choix. Et donc finalement j'appellerai quand même ça appel parce que c'est plus clair que ces histoires de marqueurs (par analogie avec l'assembleur ça revient à faire des appels avec push/pop/jmp plutôt que call/return), et je le mettrai effectivement dans les deux actions de sauts, choisis et automatique et pas dans de nouvelles actions.

Et donc ça donnerait:

```
Sauter vers: [ Statut 1]
Statut à appeler pendant le saut : [aucun]
```

Ici on saut bêtement vers statut1 comme avant.

```
Sauter vers: [ Statut 1 ]
Statut à appeler pendant le saut : [ Statut2 ]
```

Ici on pousse "Statut 1" sur la pile, et on saute vers Statut2,

```
Sauter vers: [ Dernier saut avant l'appel ]
Appeler statut pendant : [ aucun ]
```

On saut vers le statut qui est sur la pile, si la pile est vide "erreur".

```
Sauter vers: [ Dernier saut avant l'appel ]
Statut à appeler pendant le saut : [ Statut2 ]
```

On saute au statut 2 sans toucher à la pile (appel récursif en fin de fonction ou 'tail call').

#7 - 26 mai 2017 05:45 - Pierre Cros

Le vendredi 26 mai 2017 à 00:38 +0200, redmine@entrouvert.com a écrit :

> Ici il n'y a que deux statuts celui du marqueur et celui vers lequel on saute, on va forcément revenir au marqueur et on ne saura pas si l'appel a eu lieu ou pas,

Niveau fonctionnel, si on a besoin de le savoir (j'ai pas de cas d'usage), on peut jouer avec la criticité (je sais c'est pas sémantique).

Mais je pense comprendre que tu parles uniquement d'un point de vue technique.

#8 - 26 mai 2017 07:28 - Frédéric Péters

> Niveau fonctionnel, si on a besoin de le savoir (j'ai pas de cas d'usage), on peut jouer avec la criticité (je sais c'est pas sémantique).

C'est le « on sera obligé de créer une variable par exemple » du commentaire de Benjamin (que ça soit écrire dans un champ de traitement ou marquer ça dans une criticité, variable du pauvre).

Maintenant, ça m'intéresse de voir si tu comprends la proposition de Benjamin, qu'on l'amende dans son vocabulaire ou son fonctionnement, qu'on n'écrive pas quelque chose de trop compliqué.

#9 - 26 mai 2017 07:47 - Pierre Cros

Le vendredi 26 mai 2017 à 07:28 +0200, redmine@entrouvert.com a écrit :

> Maintenant, ça m'intéresse de voir si tu comprends la proposition de Benjamin, qu'on l'amende dans son vocabulaire ou son fonctionnement, qu'on n'écrive pas quelque chose de trop compliqué.

Je pense comprendre (même si éviter de parler de pile est une bonne chose) la puissance de gérer un "historique" des statuts pour une action. Mais j'ai quand même besoin, pour être en mesure de l'expliquer, de rattacher ça a du fonctionnel : "de quoi cela nous guerit-il ?"

Parce que moi "forcément revenir au marqueur" pour reprendre le vocabulaire de Benj, c'est le seul truc dont j'ai besoin et je peine donc à voir l'intérêt du reste.

#10 - 26 mai 2017 09:37 - Frédéric Péters

Je regarde le code pour reprendre et ça commence par le test qui a été conçu sur le modèle de la demande initiale (revenir au statut précédent) et je

me demande :

```
Sauter vers: [ ... ]
Statut à appeler pendant le saut : [...]
```

Comment les remplir pour assurer ça ?

Pour prendre un exemple pratique, série de statuts et puis "sur le côté" un statut "intervention du chef", vers lequel on peut aller depuis différents endroits, qui permet au chef de laisser un commentaire, qui revient ensuite là où on était.

```
Sauter vers: [ (le statut actuel) ]
Statut à appeler pendant le saut : [ Intervention du chef ]
```

Et dans "Intervention du chef" par contre, pour faire "précédent", c'est clair :

```
Sauter vers: [ Dernier saut avant l'appel ]
Statut à appeler pendant le saut : [ aucun ]
```

~~

Pour un exemple sur l'utilité de séparer ainsi, plutôt que juste avoir un marqueur, sur un exemple proche et minimal, l'agent voulant faire intervenir un chef :

- statut de traitement
  - dire que ça peut continuer (saut vers "statut suite")
  - faire intervenir un chef ("appel" vers "intervention du chef")

Avec un marqueur, après l'intervention du chef, on revient sur ce statut, on est obligé de le commencer par un saut conditionnel :

- statut de traitement
  - passer à la suite si le chef est intervenu (saut automatique sur condition)
  - dire que ça peut continuer (saut vers "statut suite")
  - faire intervenir un chef ("appel" vers "intervention du chef")

Avec le retour programmable différemment, on peut garder :

- statut de traitement
  - dire que ça peut continuer (saut vers la "statut suite")
  - faire intervenir un chef ("appel" vers "intervention du chef", "retour" à faire sur "statut suite")

#11 - 26 mai 2017 11:52 - Pierre Cros

Je pense qu'il va falloir bientôt arrêter de dire que les WF peuvent être faits par des non informaticiens... Ça complexifie sensiblement la compréhension (par rapport à un simple A/R) et je me suis déjà fait un peu insulté pendant la formation à Lyon concernant la facilité/lisiblité des actions de WF (on offre tellement de choses...). Mais pour discuter de ce dernier point c'est plutôt ici #3405

#12 - 26 mai 2017 17:31 - Thomas Noël

Frédéric Péters a écrit :

> Avec un marqueur, après l'intervention du chef, on revient sur ce statut, on est obligé de le commencer par un saut conditionnel :
>
> - statut de traitement
> - passer à la suite si le chef est intervenu (saut automatique sur condition)
> - dire que ça peut continuer (saut vers "statut suite")
> - faire intervenir un chef ("appel" vers "intervention du chef")

Je pense que même pas besoin du saut conditionnel.

Je suis agent, je vois la demande, avec deux bouton "traitement normal" / "demander au chef"
Je ne sais que faire de la demande à traiter, je clique sur "demander au chef".
Le chef gère sa partie de sous-workflow... et à un moment on lui présente un bouton "Retour" qui revient sur le statut d'appel.
Et je suis agent, la demande est revenue sur le statut du départ, j'ai toujours les deux boutons "demander au chef" / "traitement normal"
Je vois dans l'historique ce que le chef a dit de la demande, si c'est ok on continue le traitement, sinon on repart chez le chef.

> Avec le retour programmable différemment, on peut garder :
>
> - statut de traitement
> - dire que ça peut continuer (saut vers la "statut suite")
> - faire intervenir un chef ("appel" vers "intervention du chef", "retour" à faire sur "statut suite")

Pour moi, la plupart du temps, quand le tiers (ici le chef, mais ça peut être tout autre chose, y compris des mails, des appels ws) aura fini son sous-workflow, on reviendra vers le statut d'où partait l'appel. Et voilà.

J'ai l'impression de rater quelque chose, mais quoi ?

#13 - 27 mai 2017 00:55 - Frédéric Péters

> Pour moi, la plupart du temps, quand le tiers (ici le chef, mais ça peut être tout autre chose, y compris des mails, des appels ws) aura fini son sous-workflow, on reviendra vers le statut d'où partait l'appel. Et voilà.
>
> J'ai l'impression de rater quelque chose, mais quoi ?

Tu décris ce que permet le patch attaché. L'évolution imaginée, mais à prendre en compte si on le peut dès maintenant pour ne pas complexifier les migrations, c'est qu'on puisse vouloir revenir non pas sur le statut mais sur un autre.

#14 - 27 mai 2017 18:30 - Thomas Noël

Ok c'est clair pour moi maintenant. Ce qui l'est moins, c'est l'interface à présenter au programmeur du workflow. On pourrait, comme un appel de webservice, avoir des boutons de choix qui affichent/cachent une partie du reste de la config du saut. (parce que c'est déjà par exemple difficile d'expliquer qu'on ne peut pas mixer expiration et condition sur un saut automatique qui boucle sur le meme statut).

Genre :

 passer directement vers un statut cible
 aller vers le statut cible et revenir ensuite sur le statut actuel
 aller vers le statut cible et passer ensuite vers un autre statut

En allant plus loin, quid de proposer plusieurs statuts au niveau du retour ? Genre une liste de statuts nommés "accepté" / "refusé" / "spam" qui seraient à rendre disponible dans l'action de retour. Parce qu'on imagine le cas "demander l'avis à un tiers" où on aimerait que, en cas d'avis défavorable, la demande passe directement en statut "refusée". On reste ici dans l'idée de ne pas stocker une variable "décision du tiers" qui serait à analyser au retour.

#15 - 27 mai 2017 18:42 - Frédéric Péters

- Statut changé de En cours à Nouveau

- Patch proposed changé de Oui à Non

> Parce que moi "forcément revenir au marqueur" pour reprendre le vocabulaire de Benj, c'est le seul truc dont j'ai besoin et je peine donc à voir l'intérêt du reste.

Ticket créé, #16524, pour vivre dans le cadre de cette exigence fonctionnelle, sans aller au-delà.

#16 - 28 mai 2017 18:53 - Benjamin Dauvergne

Je suis tout à fait d'accord pour ne pas bloquer le besoin de Pierre sur mes idées plus lointaines. Pour moi la simple existence de ce système de piles de statuts permet d'avoir plus tard des choses plus évoluées.

Aussi je suis d'avis de ne pas trop surcharger l'action de saut et pour les besoin plus complexes d'effectivement créer une action particulière, nommée Appel ou autre.

Thomas Noël a écrit :

> Ok c'est clair pour moi maintenant. Ce qui l'est moins, c'est l'interface à présenter au programmeur du workflow. On pourrait, comme un appel de webservice, avoir des boutons de choix qui affichent/cachent une partie du reste de la config du saut. (parce que c'est déjà par exemple difficile d'expliquer qu'on ne peut pas mixer expiration et condition sur un saut automatique qui boucle sur le meme statut).
>
> Genre :
>
>
> passer directement vers un statut cible
> aller vers le statut cible et revenir ensuite sur le statut actuel
> aller vers le statut cible et passer ensuite vers un autre statut
>
> En allant plus loin, quid de proposer plusieurs statuts au niveau du retour ? Genre une liste de statuts nommés "accepté" / "refusé" / "spam" qui seraient à rendre disponible dans l'action de retour. Parce qu'on imagine le cas "demander l'avis à un tiers" où on aimerait que, en cas d'avis défavorable, la demande passe directement en statut "refusée". On reste ici dans l'idée de ne pas stocker une variable "décision du tiers" qui serait à analyser au retour.

Oui mais ça devient de plus en plus confus, déjà pour définir l'interface. Ensuite pour s'assurer que dans le workflow qui est appelé on prend bien en compte tous les retours vraiment possibles, il faudrait des analyses statiques du workflow pour valider ça (plus ou moins celles que je préconisais dans la description du ticket). Et au niveau implémentation au lieu d'avoir une pile de statuts on se retrouve avec un truc zarbi genre une pile de dico

ou un dico de piles dont la sémantique n'est plus du tout évidente.

Aussi dans mon idée même si la description n'en parle pas, l'action appel/retour devenait particulièrement utile avec des sous-workflows partageables (on définit un workflow 'intervention du chef' sans l'accrocher à aucun formulaire et on peut l'appeler de partout). Aussi on aurait pas forcément à avoir de saut explicite pour le retour, on pourrait considérer que d'atteindre un statut terminal dans ce workflow revient à remonter dans la pile si celle-ci n'est pas vide, ça pourrait peut-être même être un principe général.

#17 - 09 juin 2017 13:46 - Frédéric Péters

- Lié à Development #16524: Possibilité de saut "en arrière" ajouté

#18 - 15 novembre 2021 17:23 - Thomas Noël

- Statut changé de Nouveau à Fermé

- Planning mis à Non

implémenté par « saut avec marquage + saut vers la dernière marque » : #16524

Fichiers

0001-workflows-add-possibility-to-mark-and-jump-to-marked.patch	12,4 ko	25 mai 2017	Frédéric Péters