

Автоматизация рефакторинга скриптов рабочих процессов в YouTrack

выполнил: Голышев Роман
руководитель: Науменко Евгений, JetBrains

СПбАУ РАН

21 февраля 2018 г.

Пример бага (issue), записанного в YouTrack

Created by Heiko Rupp 05 Oct 2009 18:24 Updated by Dmitriy Smirnov 13 Feb 2018 16:22

IDEA-24944 Git SSH - Does not allow to specify SSH key in the UI

Relates to: [IDEA-166670](#) [IDEA-162666](#) [IDEA-70326](#) [IDEA-74529](#)

Depends on: [IDEA-61321](#)

Is duplicated by: [IDEA-74529](#) [OC-6092](#) [IDEA-105369](#) [IDEA-172330](#)

I am using a remote git repo that I need to access over ssh. I do have a special ssh key, that on command line gets selected when I do "git clone ssh://user@host/...git".

Maia always tries to select the default ssh key, which fails with that repo.

Maia should allow to select a ssh key to use with a certain repo.

Comments ¹²

[History](#)

[Linked Issues](#) ⁸

[Similar Issues](#)

[Time Tracking](#)



SW Sascha Weinreuter 05 Oct 2009 18:31

It doesn't allow a direct keyfile-selection, but it reads the file \$HOME/.ssh/config which allows a per-host configuration of certain ssh options. As a workaround you can configure a key for a specific host like this:

```
Host foo.xyz.net
```

```
IdentityFile ../path/key.id_dsa
```



IB Irina Petrovskaya 16 Feb 2012 13:08

the issue is still actual: there is no possibility to select the identity file from IDEA.

The ~/.ssh/config is read (but on windows instead of the set \$HOME the standard user home is used to locate .ssh folder)

Project	IntelliJ IDEA
Priority	Normal N
Type	Usability Problem
State	To be discussed
Assignee	Kirill Likhodedov
Subsystem	Version Control. Git
Fixed in builds	No Fixed in build
Fix versions	No Fix versions
Affected versions	No Affected versions
Spent time	?
Tester	No tester
Verified	No
Triaged	Yes

► Boards

Введение в предметную область, постановка проблемы

Баг-трекер YouTrack позволяет писать скрипты (workflow scripts) для расширения функциональности и автоматизации рутинных задач в процессе разработки программного обеспечения. В данный момент поддерживается два формата таких скриптов:

- ▶ Классический, основанный на MPS с редактором в виде отдельного приложения
- ▶ Экспериментальный, в виде кода на языке javascript (ES5)

Введение в предметную область, постановка проблемы

Баг-трекер YouTrack позволяет писать скрипты (workflow scripts) для расширения функциональности и автоматизации рутинных задач в процессе разработки программного обеспечения. В данный момент поддерживается два формата таких скриптов:

- ▶ Классический, основанный на MPS с редактором в виде отдельного приложения
- ▶ Экспериментальный, в виде кода на языке javascript (ES5)

Классический формат был объявлен устаревшим (deprecated). Проблемой является то, что нет удобного и простого способа для перевода скриптов из старого формата в новый.

Пример workflow с использованием старого API

```
1 rule watch if lots of votes
2
3 when votes == 10 {
4     project.leader.watchIssue(issue);
5     Assigned developer.required("Assignee required");
6     Assigned developer.notify("New issue", "You are set on this issue");
7 }
```

Эквивалентный workflow с использованием нового API

```
1 exports.rule = entities.Issue.onChange({
2   title: 'watch if lots of votes',
3   guard: function (ctx) {
4     return ctx.issue.votes === 10;
5   },
6   action: function (ctx) {
7     ctx.issue.project.leader.watchIssue(ctx.issue);
8     ctx.issue.fields.required(
9       ctx['Assigned developer'],
10      'Assignee required'
11    );
12    ctx.issue.fields['Assigned developer'].notify(
13      'New issue',
14      'You are set on this issue'
15    );
16  },
17  requirements: {...},
18 });
```

Существующие решения и их недостатки

Автоматизированных решений для трансляции скриптов **не существует**, поэтому пользователям приходится делать трансляцию вручную.

Почему это плохо:

1. Нет руководств по выполнению данной задачи \Rightarrow при ручном переводе высок шанс внесения ошибок в скрипт
2. Для перевода нужно хорошо знать и помнить старый и новый API, что затруднительно
3. Ручной перевод — неэффективный и непродуктивный труд

В результате большинство пользователей предпочитают не трогать старые скрипты так долго, как это возможно.

Цели и задачи

Цель

Разработать библиотеку/web-приложение для трансляции скриптов из старого формата в новый.

Цели и задачи

Цель

Разработать библиотеку/web-приложение для трансляции скриптов из старого формата в новый.

Задачи

- ▶ Установить однозначное соответствие между старым и новым workflow API

Цели и задачи

Цель

Разработать библиотеку/web-приложение для трансляции скриптов из старого формата в новый.

Задачи

- ▶ Установить однозначное соответствие между старым и новым workflow API
- ▶ Изучить принцип организации DSL на платформе MPS

Цели и задачи

Цель

Разработать библиотеку/web-приложение для трансляции скриптов из старого формата в новый.

Задачи

- ▶ Установить однозначное соответствие между старым и новым workflow API
- ▶ Изучить принцип организации DSL на платформе MPS
- ▶ Разработать подход для трансляции скриптов из старого формата в новый с учетом особенностей хранения скриптов в старом формате

Цели и задачи

Цель

Разработать библиотеку/web-приложение для трансляции скриптов из старого формата в новый.

Задачи

- ▶ Установить однозначное соответствие между старым и новым workflow API
- ▶ Изучить принцип организации DSL на платформе MPS
- ▶ Разработать подход для трансляции скриптов из старого формата в новый с учетом особенностей хранения скриптов в старом формате
- ▶ Реализовать непосредственно конвертер в соответствии с индустриальными стандартами программирования

Решение поставленных задач

Общий подход

- ▶ Использование системы MPS в качестве кодогенератора для трансляции *workflow*
- ▶ Генератор запускается программно с использованием ant-скриптов в отдельном процессе

Решение поставленных задач

Общий подход

- ▶ Использование системы MPS в качестве кодогенератора для трансляции workflow
- ▶ Генератор запускается программно с использованием ant-скриптов в отдельном процессе

Возникшие технические проблемы и их решение

- ▶ Валидная трансляция невозможна без дополнительной информации о проекте, в котором используется воркфлоу
- ▶ Между старой и новой версиями workflow API нет взаимно-однозначного соответствия
- ▶ На workflow editor и его внутреннюю архитектуру нет какой-либо документации, поэтому пришлось применять reverse-engineering

workflow, полученный с помощью генератора

```
1 exports.rule = entities.Issue.onChange({
2   title: 'watch if lots of votes',
3   guard: function (ctx) {
4     return (ctx.issue['votes'] === 10);
5   },
6   action: function (ctx) {
7     ctx.issue['project']['leader'].watchIssue(ctx.issue);
8     ctx.issue.fields.required(
9       ctx['Assigned developer'],
10      'Assignee required'
11    );
12    ctx.issue.fields['Assigned developer'].notify(
13      'New issue',
14      'You are set on this issue'
15    );
16  },
17  requirements: {...},
18 });
```

Результаты и выводы

Были выполнены следующие задачи:

- ▶ Установлено соответствие для большей части существующих API
- ▶ Изучена платформа MPS, а также детали реализации workflow editor
- ▶ Разработан и реализован подход для трансляции скриптов из старого формата в новый

Результаты и выводы

Были выполнены следующие задачи:

- ▶ Установлено соответствие для большей части существующих API
- ▶ Изучена платформа MPS, а также детали реализации workflow editor
- ▶ Разработан и реализован подход для трансляции скриптов из старого формата в новый

В ходе выполнения работы были получены следующие результаты:

- ▶ Генератор, основанный на MPS, способный производить трансляцию между старым и новым форматами workflow с учетом большей части существующих API
- ▶ Прототип web-приложения, позволяющего выполнять трансляцию скриптов без развертывания генератора локально