

DrupalCon Barcelona 2007

Workflow-ng

Next Generation workflows for drupal

by Wolfgang Ziegler alias fago

Presentation overview

- About myself
- Part 1 (for siteadmins)
 - Introduction of workflow-ng
 - Simple usage example
 - Future plans
- Part 2 (for developers)
 - How can you extend it? (developers)
 - Comparison to drupal 6 actions (developers)
- Q&A

About myself

- 23 year old student from Austria
- participated in SoC 2006 (Node Profile)
- Major contributions so far
 - Node Profile (Pageroute, Nodefamily, Views Fusion, ..)
 - Automatic Nodetitle
 - Fieldgroup (CCK)
 - Content Access
 - Workflow-ng

Work

employed by

- zites Internet Solutions (zites.net)
- ProKarriere (prokarriere.at)

University

- Started working on workflow-ng as a practical work for university

Thanks to the

- Information & Software Engineering Group of the Technical University of Vienna

For what workflow-ng ?

- building new flexible & powerful workflows
- customizing built-in workflows
- inspired by the existing workflow module

How does it work?

- event driven
- rule based
- action evaluation

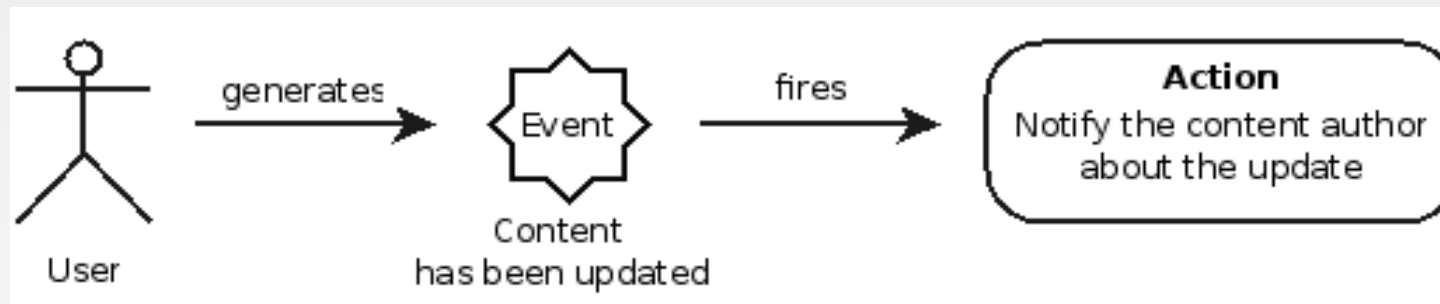
?!?

events...

- user is generating events through his actions
 - a user has logged in
 - a content has been viewed
 - a content has been edited
 - a comment has been created
- React on this events!

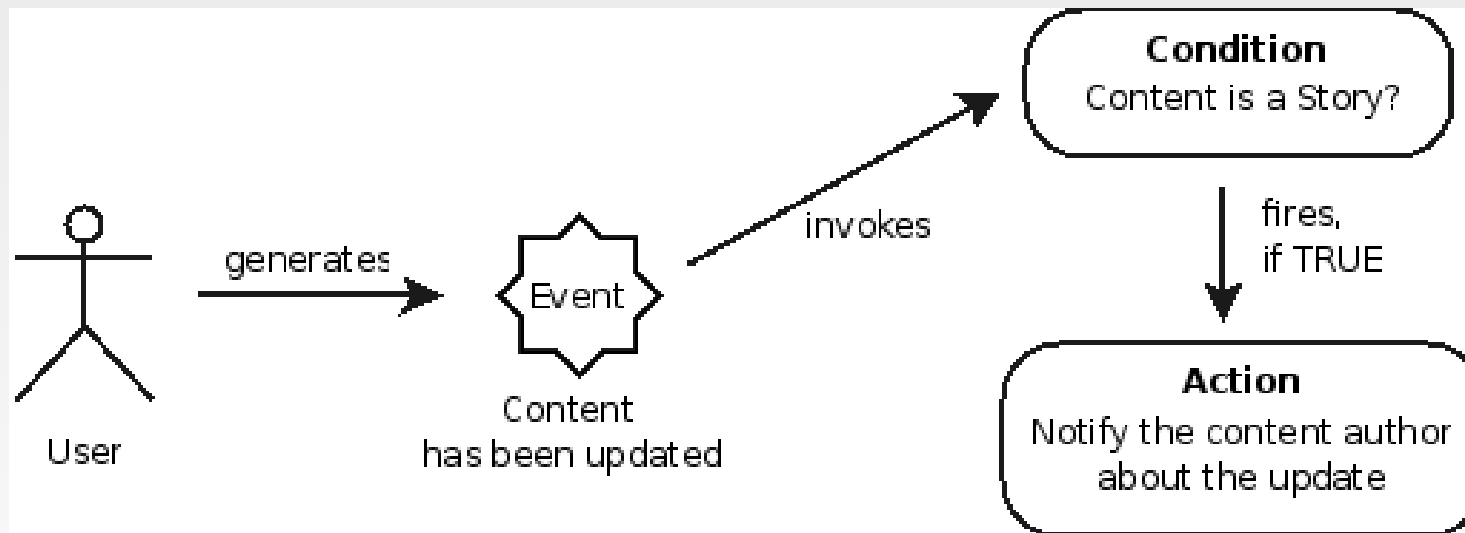
actions...

- React on the event by doing something!
 - send a mail to a user
 - publish content
 - delete a comment
 - ...



conditions...

- conditionally execute actions
 - content is published?
 - content type is a story?
 - user is an admin?



available actions/conditions?

- Actions/conditions work with some specified arguments, e.g.
 - with a content
 - with a user
 - with a content and a user (set content author)
 - whatever specified...
- Actions/conditions work with every event for which the arguments are available!

rule based?

- Configurations:
Configured conditions & actions for a given event!
- Configurations \leftrightarrow Rules
- When an event occurs,
all associated configurations are
evaluated!

Modular + Import/Export

- Modules may provide further
 - events
 - conditions
 - actions
- Modules may also provide default configurations!
- Import / Export
(Share configurations, Staging)

Simple usage example

- Redirect users on login to their account page (user/1)
 - Install workflow-ng
 - Go to its administration screen

Simple usage example

[Home](#) > [Administer](#) > [Site building](#)

Configurations overview

Overview

[Add a new configuration](#)

[Import](#)

[Export](#)

Filter by module:

Active

| Configuration | Event | Module | Status | Operations | |
|-----------------------------------|----------------------|-------------|--------|----------------------|------------------------|
| Check for profile | User has logged in | workflow-ng | Custom | Edit | Delete |
| Publish Content | Scheduled Publishing | workflow-ng | Custom | Edit | Delete |

Inactive

| Configuration | Event | Module | Status | Operations | |
|--|-------------------------------|----------------|----------|----------------------|------------------------------------|
| Print the content title of pages and stories | Content is going to be viewed | workflow-ng-ui | Modified | Edit | Revert to defaults |

Simple usage example

Home > Administer > Site building

Workflow-ng

[Overview](#) **[Add a new configuration](#)** [Import](#) [Export](#)

Event: *

User has logged in

Select on which event you want to evaluate conditions and actions.

Label: *

Redirect user to the account page on login

Choose an appropriate label for this configuration.

Active

If the configuration should be evaluated, when the associated event occurs.

Simple usage example

Home › Administer › Site building › Workflow-ng

Redirect user to the account page on login

The configuration *Redirect user to the account page on login* has been added.

Invoked on event: User has logged in

Conditions

+ Add a condition

Actions

+ Add an action

Simple usage example

Home > Administer > Site building > Workflow-ng > Redirect user to the account page on login

Add

Select an action to add: *

Page redirect

Add

Simple usage example

Home > Administer > Site building > Workflow-ng > Redirect user to the account page on login

Editing action *Page redirect*

Label: *

Choose an appropriate label for this action.

Path: *

The path, to which the user is to be redirected.

Replacement patterns

You can make use of the replacement patterns of all available arguments.

▾ Replacement patterns for logged in user

| Token | Replacement value |
|------------------------------|---|
| User tokens | |
| [account:user] | User's name |
| [account:uid] | User's ID |
| [account:mail] | User's email address |
| [account:reg-date] | User's registration date |
| [account:reg-since] | Days since the user registered |
| [account:log-date] | User's last login date |
| [account:log-since] | Days since the user's last login |
| [account:date-in-tz] | The current date in the user's timezone |
| [account:account-url] | The URL of the user's profile page. |

Simple usage example

Home > Administer > Site building > Workflow-ng > Redirect user to the account page on login

Editing action *Page redirect*

Label: *

Choose an appropriate label for this action.

Path: *

The path, to which the user is to be redirected.

Replacement patterns

You can make use of the replacement patterns of all available arguments.

—▷ [Replacement patterns for logged in user](#)

Force redirecting to the given path, even if a destination parameter is given

Per default drupal doesn't redirect to the given path, if a destination parameter is set. Instead it redirects to the given destination parameter. Most times, the destination parameter is set by appending it to the URL, e.g.

`http://example.com/user/login?destination=node/2`

Weight:

Adjust the weight to customize the ordering of actions.

Simple usage example

Home > Administer > Site building > Workflow-ng

Redirect user to the account page on login

The action *Page redirect* has been saved.

Invoked on event: User has logged in

Conditions

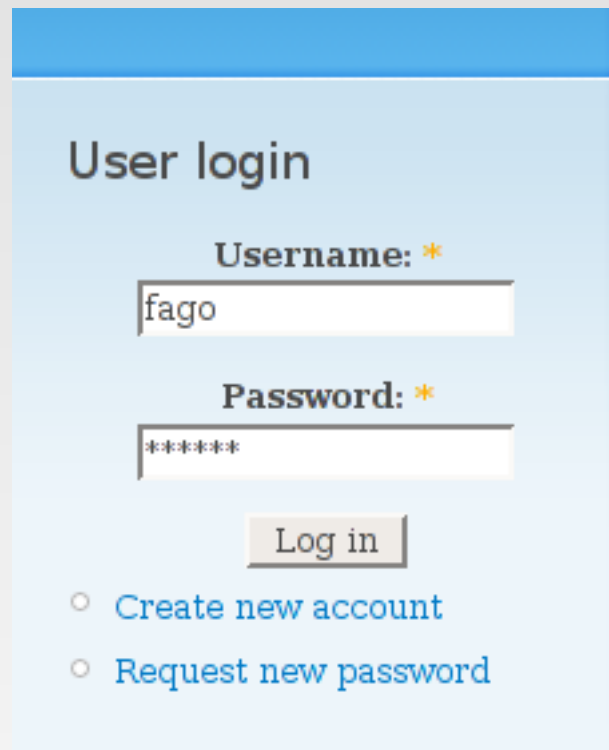
[+ Add a condition](#)

Actions

[📄 Page redirect](#)

[+ Add an action](#)

Simple usage example



A user login form with a blue header bar. The form is titled "User login" and contains two input fields: "Username: *" with the value "fago" and "Password: *" with the value "*****". Below the fields is a "Log in" button and two links: "Create new account" and "Request new password".

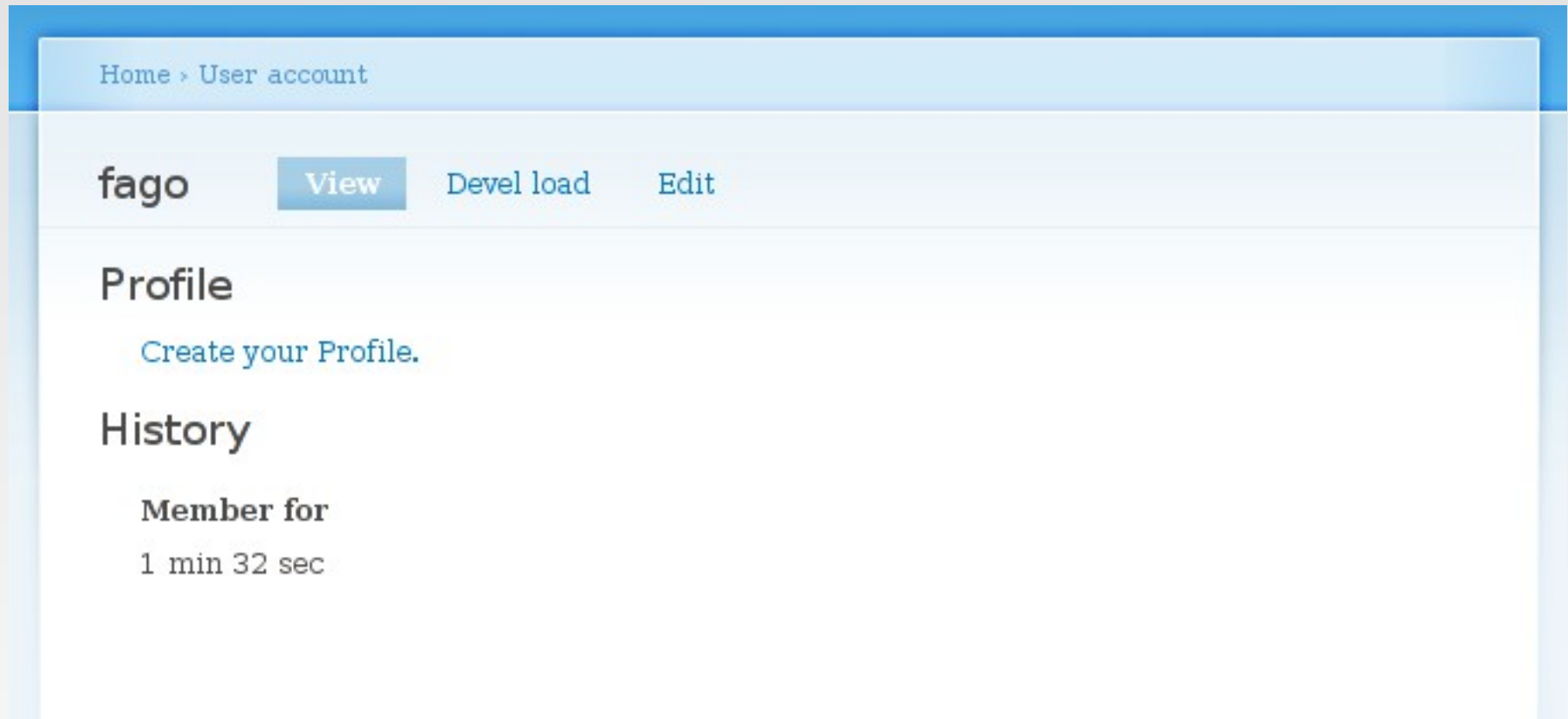
User login

Username: *

Password: *

- [Create new account](#)
- [Request new password](#)

Simple usage example



Condition groups...

Home > Administer > Site building > Workflow-ng

Print the content title of pages and stories

Invoked on event: Content is going to be viewed

Description

Note that if drupal's page cache is enabled, this event won't be generated for pages served from cache.

Conditions

OR group



OR



[+ Add another condition to this group](#)

[✂ Edit this condition group](#)

[+ Add a condition](#)

Actions

[📄 Show the title of this content](#)

[+ Add an action](#)

But how to build workflows?

- CCK, Views, Workflow-ng
- CCK field controlling the content's status (needs work, needs review, published)
- For editors:
 - View, that lists content, which needs work
 - Notify author about a review
- For reviewer:
 - View, that lists content, which needs review
- Tutorial: <http://drupal.org/node/156317>

Future plans

- port to 6.x
- more actions -> more possibilities, e.g.
 - action to control content access
 - action to update CCK fields
 - action to create content
- CCK State field
 - define possible states
 - define access control for state transitions
- Work with referenced users & nodes

Help wanted!

- Report bugs!
- Suggest new useful features!
- Developers, provide further actions, conditions and events for your modules!
- Help improving workflow-ng itself!

Part 2: for developers..

- The API allows modules to provide
 - default configurations
 - actions, which may be configurable
 - conditions, which may also be configurable
 - new events, for which conditions and actions can be configured

Example: buddylist 5.x-2.x

- Workflow-ng integration
 - Generates events, e.g. when a buddy adds another buddy
- Uses workflow-ng for notifications
 - Provides notifications as default configurations
 - Texts can be modified through workflow-ng
 - Further replacements are available (token!)
 - Easily change how users are notified (e.g. use the pm module instead of mails)

Example: buddylist 5.x-2.x

- More advantages:
 - provides the flexibility social networking sites require
 - saves a lot of code for often required functionality like sending notifications (better code reuse!)
 - extend it through further actions, e.g.:
 - add one buddy per default
 - adjust content access for buddies
 - log buddylist events
 - ..

About the API (1/2)

- multiple arguments for actions as well as conditions
- dynamic loading of needed arguments (content author, unchanged content..)
- condition evaluation with support for logical operations like OR and AND
- nesting of logical operations

About the API (2/2)

- negating conditions or logical operations
- ordered execution of conditions and actions, controlled by weights
- form API style configuration and processing of conditions and actions
- Documentation at <http://drupal.org/node/156299>

Workflow-ng and entities

- Action and condition assignment is based on arguments, which have different entity types like "node", "user", "comment"..
- Action assignment:
Workflow-ng maps available arguments from the event to the needed arguments of an action or condition!
- Workflow-ng handles saving of entities (a two times updated node is saved only once)
- One can introduce a new entity type

Expose data to workflow-ng

- Load the data in nodes/users and provide token integration
- Specify your own data structure as a new entity type and provide token integration
 - Make your entity type available through
 - new events
 - new arguments for existing events through argument loading handlers

A new entity type...

- How to work with the entity type?
 - Provide token integration
 - Provide conditions that operate on the entity type
 - Provide actions, e.g. to alter your entity!

drupal 6 actions..

- Drupal 6: Actions in Core!
- Workflow-ng is similar to drupal 6 actions
- So what are the differences?

Features..

- Workflow-ng supports
 - conditions
 - multiple arguments for actions as well as conditions
 - dynamic loading of needed arguments
 - support for logical operations like OR and AND
 - ordered execution of conditions and actions, controlled by weights

Action assignment

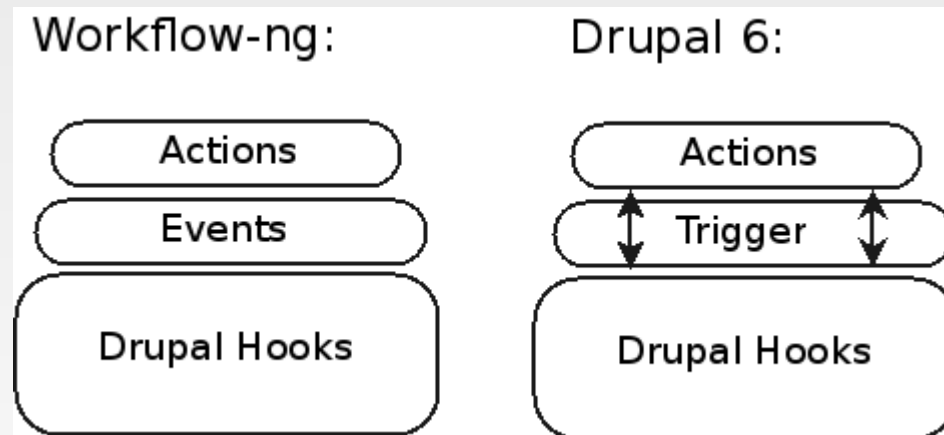
- Drupal 6 actions:
 - specify with which hooks actions can work
 - everyone can add new hooks and add compatible actions
- Workflow-ng
 - action specifies needed arguments
 - event specifies available arguments
 - actions can be configured for every event, which provides the necessary arguments!

events & triggers...

- Workflow-ng uses events to abstract hooks whereas drupal 6 uses triggers
- Triggers don't fully abstract hooks, they only unify the context for hook_nodeapi, hook_user and hook_comment

events & triggers...

- Apart from that actions have to work with the data provided by hooks
- but hooks change, so the data changes...



events & triggers...

- drupal 6 actions:
 - Actions need to be able to unify the context of any further supported hook!
- Workflow-ng:
 - Unification of the context is done once by the event generation code
 - Each module may introduce new events, which work with existing actions

hooks & possible problems...

- Consider a new module A offering a new hook, and another module B offering a new action
- Module A specifies the hook and the compatible actions it knows
- Module B specifies the action and the compatible hooks it knows
- Module A and B don't know each other, so hook and action can't be combined

Compatibility...

- Drupal 6 actions and workflow-ng actions are **not** compatible!
- Big parts of the action system work equally, but:
 - the action metadata looks different, as workflow-ng requires other metadata (arguments)
 - workflow-ng cleanly supports more than one argument per action
 - entity saving works different

Possible solution .. ?

- Write actions, which work with both!?
 - That's bad as action writing gets more complicated!
- Support drupal 6 actions with workflowing!
 - planned for the 6.x port!
- Get the best of both into drupal 7 :-)

Q & A

- Any questions?