Lesson 4 - Adding Triggers

Triggers are actions executed in certain events. For example, sending an email notification when a page is approved, changing the state of a page when a label is added, or removing view restrictions when a page is published.

In this tutorial, we take the workflow we created in the previous tutorial and add email notifications and on-screen messages.

- Adding "Under Review" when the page is submitted for review
- Setting a message when pages are created
- Sending an email notification when a page is ready for final approval
- Adding a 'reject' email when Editor-in-Chief rejects the page
- Finalizing after the page is published, and manipulating labels
- Final Workflow

Adding "Under Review" when the page is submitted for review

We want to set the page message to Under review whenever the page is submitted for review.

We need to create a trigger for the statechanged event:

```plaintext
{workflow:name=Basic Workflow}
...
{trigger:statechanged|state=Ready}
  {set-message:style=info}Under review{set-message}
{trigger}
{workflow}
```

The message is set using the {set-message} macro.

Now, whenever we submit the page for review, the message shown is:

Setting a message when pages are created

We want to show a message whenever a page is created. Create a trigger for the pagecreated event:

```plaintext
{workflow:name=Basic Workflow}
....
{trigger:pagecreated}
  {set-message:style=note}Draft{set-message}
{trigger}
...
{workflow}
```

Sending an email notification when a page is ready for final approval

This trigger works in a Space-wide workflow only as it handles the pagecreated event. See how you can apply a workflow to all pages in a Space.

We want to send an email notification when a page is ready for final approval.

```plaintext
{workflow:name=Basic Workflow}
....
{trigger:pagecreated}
  {set-message:style=note}Finalizing after the page is published, and manipulating labels{set-message}
{trigger}
...
{workflow}
```
Create a trigger for the statechanged event and use the {send-email} macro to send the notification:

```markdown
{workflow:name=Basic Workflow}
...
(trigger:statechanged|state=Reviewed)
  {send-email:user=0Editor in Chief0|subject=Page ready for final approval}
    The page @page@ has been approved by [-@user0] and is waiting
    for your final approval.
    {pageactivity}
  {send-email}
{trigger}
{workflow}
```

- Use the @Editor in Chief@ parameter
- Use the (page) and((user)) value references to include links to the page and current user
- Use the {pageactivity} to include the latest activity in the email body

Now, whenever the page's state is change to **Reviewed**, an email is sent to the *Editor-in-Chief*:

![Email notification example]

**Adding a 'reject' email when Editor-in-Chief rejects the page**

We want to send an email to the reviewer if a page is rejected by the Editor-in-Chief.

```markdown
{workflow:name=Basic Workflow}
...
(trigger:pagerejected|approval=Editor in Chief)
  {send-email:user=Reviewer|subject=Page rejected}
    The page @page@ has been rejected by [-@user0]
    {pageactivity}
  {send-email}
{trigger}
{workflow}
```

- We use the{user} (Review>approver) value reference to refer to the user who gave the Review approval.

**Finalizing after the page is published, and manipulating labels**

When the page's state changes to **Published**, we want to show a different message and add a label.

```markdown
...
(trigger:publish|state=Published)
  {send-email:user=Reviewer|subject=Page finalized}
    The page @page@ has been published and is ready for final review.
    {pageactivity}
  {send-email}
{trigger}
...
{workflow}
```
The \textcode{set-label} macro adds labels. We remove them using the \textcode{remove-label} macro.

To make sure nobody deletes the newly added label, use a trigger connected to the system 'labelremoved' event.

This \textcode{labelremoved} trigger is going to fire if the \textcode{released} label is removed by a user other that the one set in the Editor-in-Chief metadata value.

Our workflow is complete! All the transitions are recorded, and we can review the published version on a page:
The page @page@ has been approved by [~@user@] and is waiting for your final approval.

The page @page@ has been rejected by [~@user@]

Published page!

You are not allowed to remove this label