NISTtheDocs2Death

Jonathan E. Guyer

Aug 07, 2023

ACTIONS:

This GitHub action[1] uses the Sphinx tool to build documentation in https://github.com/usnistgov projects and then host that documentation on https://pages.nist.gov, as an approximation of ReadTheDocs.

---

[1] Certain commercial firms and trade names are identified in this document in order to specify the installation and usage procedures adequately, or to provide context. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that related products are necessarily the best available for the purpose.

USAGE

- Configure your repo for publishing on pages.nist.gov.

- For each branch of your repository where you want to host documentation, add a workflow, such as `.github/workflows/NISTtheDocs2Death.yml`:

```yaml
name: "Build Documentation"

on: [push, pull_request, delete]

jobs:
  docs:
    runs-on: ubuntu-latest
    steps:
      - uses: usnistgov/NISTtheDocs2Death@main
        with:
          docs-folder: docs/
          formats: |-
            epub
            pdf
```

1.1 NISTtheDocs2Death Action

This GitHub action uses the Sphinx tool to build documentation in https://github.com/usnistgov projects and then host on https://pages.nist.gov as an approximation of ReadTheDocs.

1.1.1 Usage

This action is invoked by adding a workflow file to your repository, such as `.github/workflows/NISTtheDocs2Death.yml`:

```yaml
name: "Build Documentation"

on: [push, pull_request, delete]

jobs:
  docs:
    runs-on: ubuntu-latest
    steps:
```

```
    - uses: usnistgov/NISTtheDocs2Death@0.1
      with:
        docs-folder: docs/
        pages-branch: 'nist-pages'
        pages-url: 'https://pages.nist.gov'
        formats: ''
        build-html-command: make html
        build-epub-command: make epub
        build-pdf-command: make epub
        pre-build-command: ''
```

1.1.2 Inputs

`docs-folder`

The folder containing your Sphinx configuration.

`pages-branch`

The branch linked to your documentation server.

`pages-url`

URL of the web server for served documentation.

`formats`

Type(s) of output desired in addition to html (`pdf`, and/or `epub`).

Note: GitHub Actions' YAML implementation does not support list or array elements, so use a multiline string to declare formats, e.g.,

```
formats: |-
  epub
  pdf
```

`build-html-command`

The command used by `usnistgov/sphinx-action` to build your html documentation.

---

build-epub-command

The command used by `usnistgov/sphinx-action` to build your ePUB documentation.

build-pdf-command

The command used by `usnistgov/sphinx-action` to build your PDF documentation.

pre-build-command

Run by `usnistgov/sphinx-action` before the build command. You can use this to install system level dependencies, for example, with "`apt-get update -y && apt-get install -y perl`".

1.1.3 Implementation

This action implements a composite workflow with the following major steps:

1. `actions/checkout`

2. *borg_the_docs Sub-Action*

3. `usnistgov/sphinx-action`

4. *update_pages Sub-Action*

5. `ad-m/github-push-action`

6. `actions/upload-artifact`

1.2 `borg_the_docs` Sub-Action

This GitHub action is invoked by the main *NISTtheDocs2Death Action* to modify the Sphinx configuration before building the documentation.

1.2.1 Usage

This action is invoked as a step of the composite workflow of the *NISTtheDocs2Death Action*. There is no reason to invoke this action yourself.

```
- name: Add cruft to theme
  id: borg-the-docs
  uses: ./../../_actions/current/borg_the_docs
  with:
    docs-folder: 'docs/'
```

Note: This action must be synchronized with the invoked version of the *NISTtheDocs2Death Action*, but "`usnistgov/ NISTtheDocs2Death/borg_the_docs@${{ github.action_ref }}`" doesn't work, hence the clumsy `uses:` statement.

1.2.2 Inputs

docs-folder

The folder containing your Sphinx configuration.

1.2.3 Outputs

borged-docs-folder

The folder containing the Sphinx configuration modified by this Action.

borged-build-folder

The folder containing the Sphinx build outputs.

1.2.4 Implementation

This action implements a Docker workflow step. The Docker ENTRYPOINT installs any requirements.txt in the Sphinx configuration directory, wraps the conf.py file with a BorgedConfFile object, and invokes assimilate_theme().

API

borg_the_docs_action

---

borg_the_docs_action

Modules

1.3 update_pages Sub-Action

This GitHub action is invoked by the main *NISTtheDocs2Death Action* to move built documentation into your designated *pages-branch*.

1.3.1 Usage

This action is invoked as a step of the composite workflow of the *NISTtheDocs2Death Action*. There is no reason to invoke this action yourself.

```
- name: Commit documentation changes
  uses: ./../../_actions/current/update_pages
  with:
    docs-folder: 'docs/'
    default-branch: 'main'
    pages-branch: 'nist-pages'
    pages-url: 'https://pages.nist.gov'
```

Note: This action must be synchronized with the invoked version of the *NISTtheDocs2Death Action*, but "usnistgov/ NISTtheDocs2Death/borg_the_docs@${{ github.action_ref }}" doesn't work, hence the clumsy `uses:` statement.

1.3.2 Inputs

`docs-folder`

The folder containing your Sphinx configuration.

`default-branch`

The default branch, as configured on GitHub.

`pages-branch`

The branch linked to your documentation server.

`pages-url`

URL of the web server for served documentation.

1.3.3 Implementation

This action implements a Docker workflow step. The Docker `ENTRYPOINT` wraps the Sphinx configuration directory in a `SphinxDocs` object, wraps the `GITHUB_REPOSITORY` in a `Repository` object, and invokes `update_pages()`.

API

update_pages_action

update_pages_action

Modules

1.4 Design

1.5 Customization

NISTtheDocs2Death uses two sets of templates that determine the appearance of the hosted documentation.

1.5.1 `borg_the_docs` templates

The first set of templates is used by the *borg_the_docs Sub-Action* to modify the Sphinx configuration.

These templates can be customized by forking this repository.

`conf.py`

This template file overlays the html_theme you chose for your documentation with the `ntd2d` theme, described next.

`ntd2d/`

This template directory provides a Sphinx theme that modifies your chosen documentation theme.

`ntd2d/layout.html`

This template file inserts standard NIST headers and footers and a dropdown menu that allows selecting different versions of your documentation.

`ntd2d/static/ntd2d.css_t`

This template file controls the appearance of the dropdown menu.

1.5.2 `update_pages` templates

The second set of tempaltes is used by the *update_pages Sub-Action* sub-action to create the pages on the hosting site that enable switching between different documentation variants.

You can customize any of these templates by copying them to a `_templates/` directory at the root of your *nist-pages* branch and editing them to suit.

These templates use Python string format syntax (Sphinx and pages.nist.gov already fight over Jekyll templating, so we're not getting fancy, here).

`downloads.html`

A section inserted into *menu.html* if a documentation variant has any downloadable output, e.g., PDF or ePUB.

Available subsitution keywords are:

- `downloads`: A pre-formatted string with each downloadable output formatted by *download_item.html*.

`download_item.html`

Formats a link to a single downloadable output.

Available subsitution keywords are:

- `href`: URL of the downloadable output.
- `kind`: Type of downloadable output, e.g., PDF or ePUB.

`index.html`

The default page for your documentation displayed at https://pages.nist.gov/*{repository}*.

Available subsitution keywords are:

- `owner`: The GitHub user or organization for your repository.
- `repository`: The name of your repository.
- `variants`: The result of filling the *variants.html* template.

`menu.html`

The dropdown menu that allows selecting different branches and tags of your documentation.

Available subsitution keywords are:

- `downloads`: The result of filling the *downloads.html* template.
- `tree_url`: The GitHub URL corresponding to the active tag or branch.
- `variant`: The name of the active tag or branch.
- `variants`: The result of filling the *variants.html* template.

`ntd2d_active.css`

Style sheet that controls the appearance of the active tag or branch in the dropdown menu.

Available subsitution keywords are:

- `variant`: The name of the active tag or branch.

`variants.html`

Lists tags and branches that are *configured* to serve documentation with this Action.

Available subsitution keywords are:

- `branches`: A pre-formatted string with each git branch formatted by *variant_item.html*.
- `latest`: A pre-formatted string with the `HEAD` of the default GitHub branch formatted by *variant_item.html*.
- `stable`: A pre-formatted string with the `stable_version` that has the highest version identifier, as formatted by *variant_item.html*.
- `stable_versions`: A pre-formatted string with the tags or branches that satisfy the **PEP 440** version specification and aren't pre-releases, each formatted by *variant_item.html*.
- `versions`: A pre-formatted string with the tags or branches that satisfy the **PEP 440** version specification, each formatted by *variant_item.html*.

`variant_item.html`

Formats a link to a single tag or branch.

Available subsitution keywords are:

- `href`: URL of the downloadable output.
- `kind`: Type of downloadale output, e.g., PDF or ePUB.

1.6 Q & A

1.6.1 Can't I just use Read the Docs?

Nope.

1.6.2 Is there a reason?

Yep.

1.6.3 Is it a good reason?

Nope.

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

b
borg_the_docs_action, 6

u
update_pages_action, 8

B
borg_the_docs_action
    module, 6

E
environment variable
    GITHUB_REPOSITORY, 7

G
GITHUB_REPOSITORY, 7

M
module
    borg_the_docs_action, 6
    update_pages_action, 8

P
Python Enhancement Proposals
    PEP 440, 10

U
update_pages_action
    module, 8