How we created a Documentation Framework that works across a group of vendors in the Sovereign Cloud Stack community.

A journey from single documentation files to a meaningful information architecture.

@maxwolfs
Knowledge Management Engineer at Sovereign Cloud Stack
What is it?

A set of github action workflows to copy markdown files and folders from many A’s to a single B. Once that’s done, B is being built into a single page application resulting in the Sovereign Cloud Stack documentation at

https://docs.scs.community
Sovereign Cloud Stack

What is it?

The Sovereign Cloud Stack combines the best of Cloud Computing in one unified standard. SCS is built, backed, and operated by an active open-source community worldwide.
The SCS Stack is made up from many modules, the docs are heavily distributed.
The Challenge

Integrators and Operators have to manage different documentations with different git repos and different docs.

It would be nice to have them within one platform!

But how?
Aim and Requirements to the documentation

- minimal ruleset where approval is *nobrainer*
- all doc files in one place
- low entry hurdle for companies with existing repositories – no one has to make any major changes

git submodules? ✗
git subtree? ✗
Unleashing community superpowers

At the SCS Hackathon in November 22 in Cologne hosted by plusserver we have developed a working Prototype of a custom workflow. 🚀
How does it work?

```plaintext
build_and_deploy.yml
on: workflow_dispatch

Matrix: synchronize-scs-docs ...

✔ synch.../provide_repos_json 4s
✔ s.../sync_repos (osism/op... 9s
✔ s.../sync_repos (Sovereign... 7s
✔ s.../sync_repos (Sovereign... 8s
✔ s.../sync_repos (Sovereign... 5s

✔ build_and_deploy 1m 39s
```
How does it work?

docusaurus as react based static site generator with some GitHub Actions

Magic 🪄

3 GitHub Actions Workflows:
- Collecting
- Distilling
- Building
Defining a package.docs.json in the root directory of the docs-page

docs.package.json

```json
[
  {
    "repo": " SovereignCloudStack/docs",
    "source": "community/contribute",
    "target": "community",
    "label": ""
  },
  {
    "repo": " SovereignCloudStack/docs",
    "source": "operations",
    "target": "docs",
    "label": ""
  },
  {
    "repo": " SovereignCloudStack/release-notes",
    "source": "*.md",
    "target": "docs",
    "label": "release-notes"
  },
  {
    "repo": " osism/openstack-image-manager",
    "source": "doc/*.md",
    "target": "docs",
    "label": "openstack-image-manager"
  }
]
```
### 1. Reading the docs.package.json and defining a matrix strategy

```yaml
provide_repos_json

runs-on: ubuntu-latest
outputs:
  matrix: ${{ steps.set-matrix.outputs.matrix }}
steps:
  - name: Checkout the docusaurus repo (B)
    uses: actions/checkout@v3
  - name: Set Matrix
    id: set-matrix
    run: |
      REPOS=$(echo $(cat ./docs.package.json) | sed 's/ //g')
      echo "::set-output name=matrix::$REPOS"
```
2. Synchronizing and distilling workflow

```bash
# Sync repos

needs: provide_repos.json
runs-on: ubuntu-latest
strategy:
  max-parallel: 1
matrix:
  repos: [${( fromJson(needs.provide_repos.json.outputs.matrix))}]

steps:
- name: checkout the docusaurus repo (B)
  uses: actions/checkout@v3
  with:
    token: ${ secrets.PAT }

- name: remove previous docs for clean install
  if: ${ matrix.repos.label != null }
  run:
    - rm -rf ${pwd}/${matrix.repos.target}/$(matrix.repos.label)

- name: remove previous docs for clean install
  if: ${ matrix.repos.label == null }
  run:
    - rm -rf ${pwd}/${matrix.repos.target}/$(matrix.repos.source)
    - rm -rf ${pwd}/${matrix.repos.source}

- name: clone repo A which is about to be synchronized
  uses: audashed/git-clone-action@v1.8.1
  with:
    repository: ${matrix.repos.repo}
    path: 'repo_to_be_edited'
```

- name: remove git folders from A
  run: |
    rm -rf $(pwd)/repo_to_be_edited/.git

- name: remove README.md files from A
  run: |
    find $(pwd)/repo_to_be_edited -name "README.md" | xargs rm -f

- name: create docusaurus subdirectory
  run: |
    mkdir $(pwd)/$(matrix.repos.target)/$(matrix.repos.label) || true

- name: copy docs content from A to B
  run: |
    cp -r $(pwd)/repo_to_be_edited/$(matrix.repos.source) $(pwd)/$(matrix.repos.target)/$(matrix.repos.label)

- name: remove repo A
  run: |
    rm -rf $(pwd)/repo_to_be_edited/

- name: commit and push B
  uses: EndBug/add-and-commit@v9
  with:
    author_name: 'bot@cs'
    committer_name: Action Bot
    commit: --signoff
    message: "distill $(matrix.repos.repo)"
    push: true
    pull: --rebase --autostash
2.1. Synchronizing and distilling workflow in Detail

```yaml
sync_repos

needs: provide_repos_json
runs-on: ubuntu-latest
strategy:
  max-parallel: 1
  matrix:
    repos: ${{ fromJson(needs.provide_repos_json.outputs.matrix) }}
steps:
  - name: checkout the docusaurus repo (B)
    uses: actions/checkout@v3
    with:
      token: ${{ secrets.PAT }}
```
2.2. Synchronizing and distilling workflow in Detail

```yaml
- name: remove previous docs for clean install
  if: ${{matrix.repos.label !== null}}
  run:
    rm -rf $(pwd)/${{matrix.repos.target}}/${{matrix.repos.label}}

- name: remove previous docs for clean install
  if: ${{matrix.repos.label == null}}
  run:
    rm -rf $(pwd)/${{matrix.repos.target}}/
    rm -rf $(pwd)/${{matrix.repos.source}}

- name: clone repo A which is about to be synchronized
  uses: sudosubin/git-clone-action@v1.0.1
  with:
    repository: ${{matrix.repos.repo}}
    path: 'repo_to_be_edited'
```
2.3. Synchronizing and distilling workflow in Detail

- name: remove git folders from A
  run: |
    rm -rf $(pwd)/repo_to_beEdited/.git

- name: remove README.md files from A
  run: |
    find $(pwd)/repo_to_beEdited -name "README.md" | xargs rm -f

- name: create docusaurus subdirectory
  run: |
    mkdir $(pwd)/${{matrix.repos.target}}/${{matrix.repos.label}} || true

- name: copy docs content from A to B
  run: |
    cp -r $(pwd)/repo_to_beEdited/${{matrix.repos.source}}
    $(pwd)/${{matrix.repos.target}}/${{matrix.repos.label}}
2.4. Synchronizing and distilling workflow in Detail

```yaml
- name: copy docs content from A to B
  run: |
    cp -r $(pwd)/repo_to_be_edited/$(matrix.repos.source) $(pwd)/$(matrix.repos.target)/$(matrix.repos.label)

- name: remove repo A
  run: |
    rm -rf $(pwd)/repo_to_be_edited/

- name: commit and push B
  uses: EndBug/add-and-commit@v9
  with:
    author_name: 'bot@scs'
    committer_name: Action Bot
    commit: --signoff
    message: "distill $(matrix.repos.repo)"
    push: true
    pull: --rebase --autostash
```
The Result: https://docs.scs.community
Feedback

- Ideas, Feedback, Critics?
- Have a look at our repository
- SIG Documention MeetUp every 2nd monday from 11:05 AM CET: Community Calendar

https://github.com/SovereignCloudStack/docs-page
What is still to come

- structure wise adapting towards diataxis
- currently triggered manually, soon to be automated
- interactive overview about the standards/stack
- fancy community space

Join our community @matrix