



DLF

NETWORKING

Virtual Technical Meetings

ONAP - Write the Docs

Jakob Krieg

1 State of Documentation in ONAP

2 Technical Overview

3 Tips & Tricks

- Convert Confluence to .rst
- Live preview in VSC
- Tabs, Videos, ... in ReadTheDocs
- Testing in ReadTheDocs
- Your Tips & Tricks

4 Conclusion

ReadTheDocs: for all formal ONAP E2E and component documentation, e.g.

ONAP overview, architecture, API, release notes

ONAP developer guides (e.g. Documentation guide)

ONAP user guides (E2E), component guides, use cases, tutorials

Confluence Wiki: for ONAP release, project, subcommittee and development related content, e.g.

- project management (meetings, plans, milestones, members, ...)
- ongoing activities and discussions
- ONAP community event documentation

State of documentation in ONAP

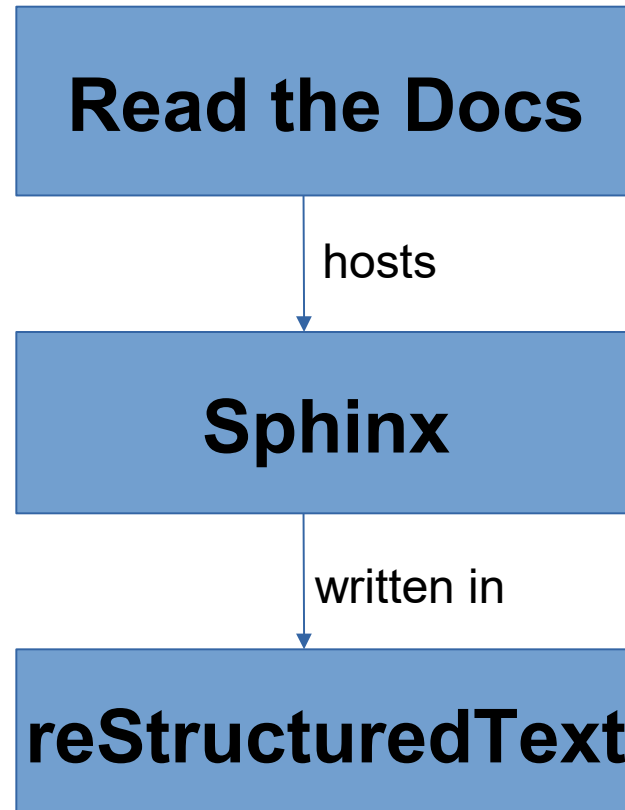
Problems:

- documentation in Confluence should be in Read the Docs (RDT)
- duplicate and inconsistent documentation
- documentation in Confluence not versioned

Goals:

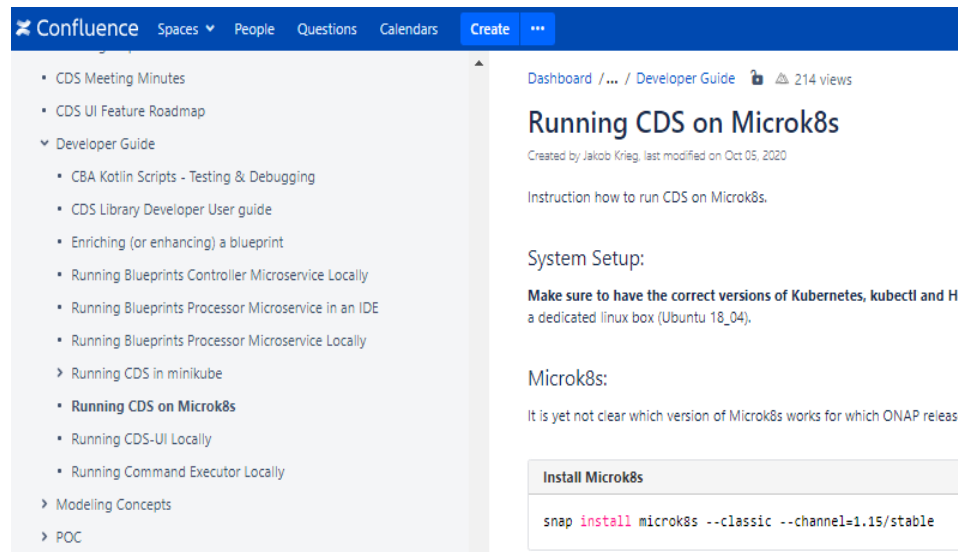
- move RDT related documentation from Confluence to RDT
- Provide same possibilities in Read the Docs
- Be efficient

Technical Overview

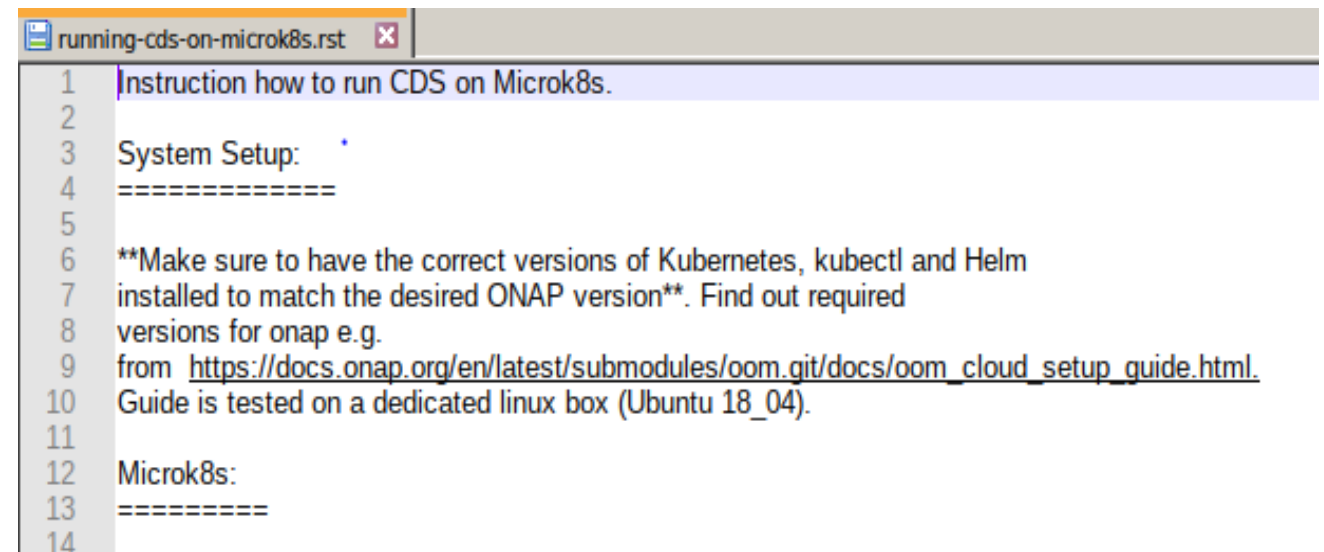


Converting Confluence pages to reStructuredText (.rst)

- Tool available to convert Confluence pages to .rst files,
- Several pages can be added to pagelist
- Not all elements in Confluence are covered



The screenshot shows a Confluence page titled "Running CDS on Microk8s" within a "Developer Guide" space. The page content includes a "System Setup" section with a warning to use correct versions of Kubernetes, kubectl, and Helm on a dedicated Linux box (Ubuntu 18_04), and a "Microk8s" section with a note that it's unclear which version works for which ONAP release. A code block for installing Microk8s is visible at the bottom: `snap install microk8s --classic --channel=1.15/stable`.



The screenshot shows a text editor window titled "running-cds-on-microk8s.rst". The content is a reStructuredText document with the following structure:

```
1 | Instruction how to run CDS on Microk8s.
2 |
3 | System Setup:
4 | =====
5 |
6 | **Make sure to have the correct versions of Kubernetes, kubectl and Helm
7 | installed to match the desired ONAP version**. Find out required
8 | versions for onap e.g.
9 | from https://docs.onap.org/en/latest/submodules/oom.git/docs/oom\_cloud\_setup\_guide.html.
10 | Guide is tested on a dedicated linux box (Ubuntu 18_04).
11 |
12 | Microk8s:
13 | =====
14 |
```

Creating Documentation with VSC Extensions

The screenshot displays the Visual Studio Code interface with a developer guide open. The guide is titled "Running Blueprints Processor Microservice in an IDE" and is part of the "Developer Guide" for the "Running Blueprints Processor Microservice in an IDE". The guide includes sections for "Objective", "Check out the code", and "Build it locally".


The "Objective" section states: "Have the processor running locally is to use the IDE to run the code, while having the database running in a container. This way, code changes can be conveniently tested and debugged."

The "Check out the code" section states: "Check out the code from Gerrit: <https://gerrit.onap.org/r/#/admin/projects/ccsdk/cds>"

The "Build it locally" section states: "In the checked out directory, type `.. code-block:: bash`"

The screenshot also shows the Explorer view on the left, the Search docs bar, and the Table of Contents for the Developer Guide.

Features: Embedding Videos, Tabbed Views



Search docs

- Modeling Concepts
- Controller Blueprints Studio Processor
- Blueprints Processor
- Developer Guide
- Installation Guide
- Design Time Tools Guide

Use Cases

- Wordpress CNF in CDS (POC)
 - Presentation of Gerald Karam (2020-09-08)
 - PNF Simulator Day-N config-assign/deploy
- CDS Designer UI

Wordpress CNF in CDS (POC)

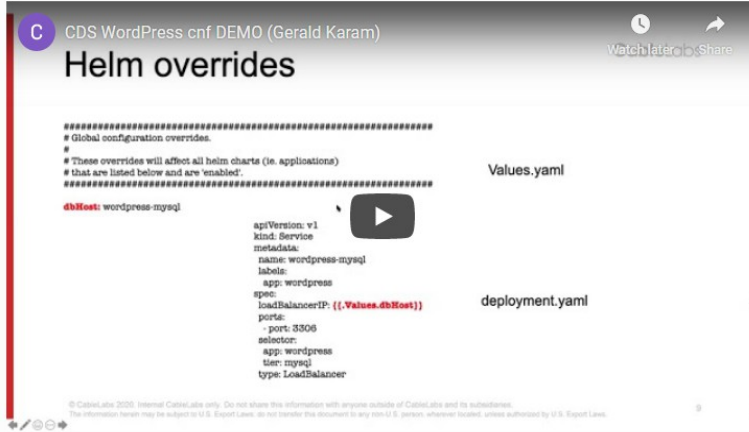
This demo by CableLabs shows an easy to use POC how to use/deploy VNFs in CDS and do resource assignment.

Detailed description will follow as soon as there is an acknowledgement from CableLabs that content can be published.

Goal is to use CDS (ONAP) in a very simple and understandable way. Azure, AWS and Kubernetes are used as VIMs through CDS.

This demo was tested on Frankfurt.

Presentation of Gerald Karam (2020-09-08)



CDS WordPress cnf DEMO (Gerald Karam)


Helm overrides

Values.yaml

```
#####  
# Global configuration overrides.  
#  
# These overrides will affect all helm charts (ie. applications)  
# that are listed below and are 'enabled'.  
#####  
dbHost: wordpress-mysql
```

deployment.yaml

```
apiVersion: v1  
kind: Service  
metadata:  
  name: wordpress-mysql  
  labels:  
    app: wordpress  
spec:  
  loadBalancerIP: {{.Values.dbHost}}  
  ports:  
  - port: 3306  
  selector:  
    app: wordpress  
    tier: mysql  
  type: LoadBalancer
```



Search docs

- Modeling Concepts
 - Controller Blueprint Archive (.cba)
 - Tosca.Meta
 - Dynamic Payload
 - Enrichment
 - External Systems support
 - Expression
 - Data Dictionary
 - Data type
 - Artifact Type
- Node type
 - Workflow
 - Template
 - Scripts
 - Southbound Interfaces
 - Tests
- Controller Blueprints Studio Processor
- Blueprints Processor
- Developer Guide
- Installation Guide
- Design Time Tools Guide
- Use Cases
- CDS Designer UI

Node type

TOSCA definition

In CDS, we have mainly two distinct types: components and source. We have some other type as well, listed in the other section.

Component Source Other

Component:

Used to represent a **functionality** along with its **contract**, such as **inputs**, **outputs**, and **attributes**

Here is the root component TOSCA node type from which other node type will derive:

```
tosca.nodes.Component  
  
{  
  "description": "This is default Component node",  
  "version": "1.0.0",  
  "derived_from": "tosca.nodes.Root"  
}
```

Bellow is a list of supported components

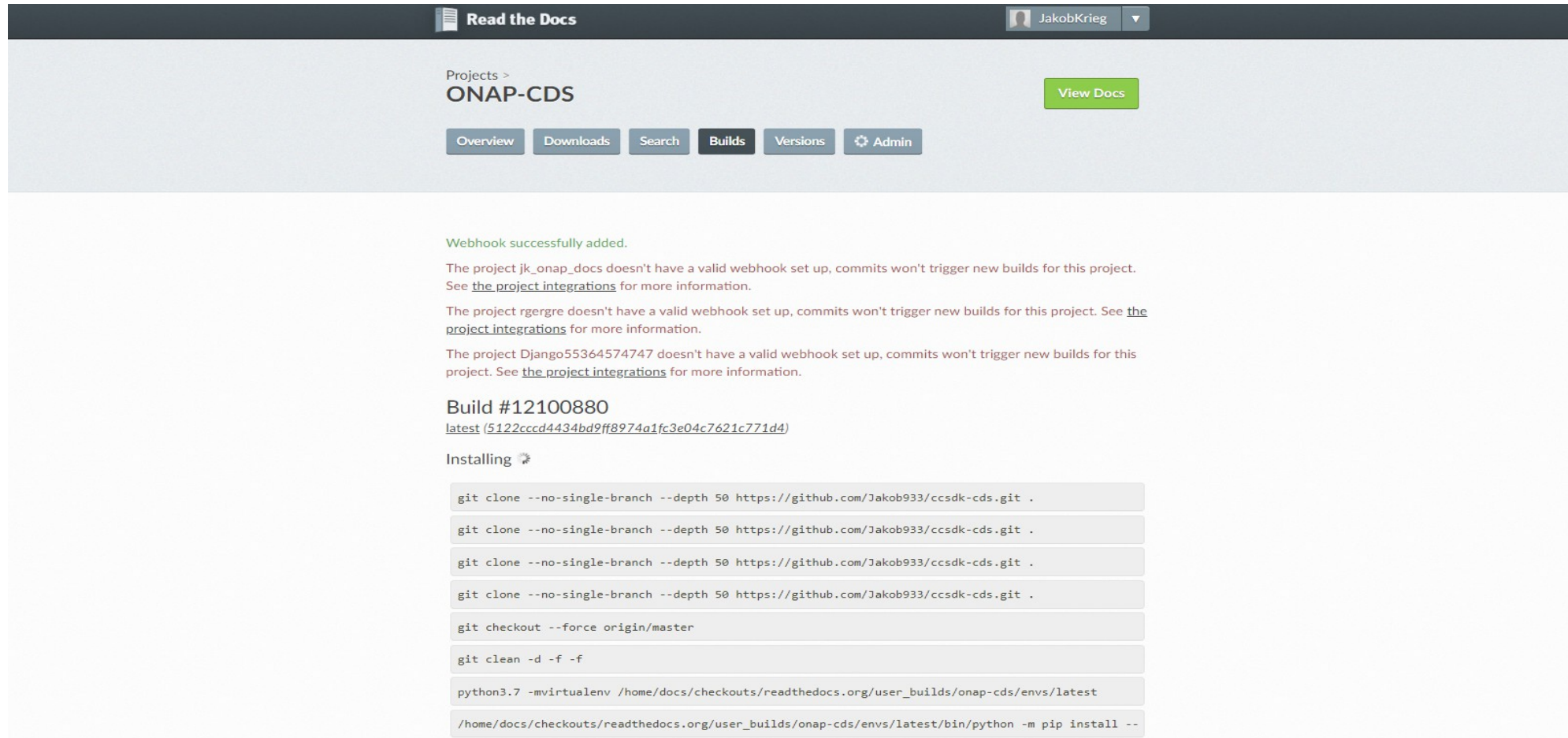
resource-resolution script-executor remote-script-executor remote-ansible-executor

component-resource-resolution:

Used to perform resolution of resources.

Requires as many as artifact-mapping-resource (see Artifact Type -> Mapping) AND artifact-template-velocity (see Artifact Type -> Jinja) as needed.

Testing ReadTheDocs



The screenshot shows the ReadTheDocs interface for the ONAP-CDS project. The header includes the site name "Read the Docs" and the user "JakobKrieg". The main content area displays the project name "ONAP-CDS" and a "View Docs" button. Below this are navigation tabs for "Overview", "Downloads", "Search", "Builds", "Versions", and "Admin". The main content area contains a message: "Webhook successfully added." followed by three paragraphs of text, each stating that a specific project (jk_onap_docs, rgergre, and Django553645747) does not have a valid webhook set up and providing a link to "the project integrations" for more information. Below this is a section for "Build #12100880" with the tag "latest (5122cccd4434bd9ff8974a1fc3e04c7621c771d4)". The "Installing" section contains a list of terminal commands for cloning the repository, checking out the master branch, cleaning the directory, and installing dependencies using pip.

Read the Docs JakobKrieg

Projects > ONAP-CDS View Docs

Overview Downloads Search Builds Versions Admin

Webhook successfully added.

The project jk_onap_docs doesn't have a valid webhook set up, commits won't trigger new builds for this project. See [the project integrations](#) for more information.

The project rgergre doesn't have a valid webhook set up, commits won't trigger new builds for this project. See [the project integrations](#) for more information.

The project Django553645747 doesn't have a valid webhook set up, commits won't trigger new builds for this project. See [the project integrations](#) for more information.

Build #12100880
latest (5122cccd4434bd9ff8974a1fc3e04c7621c771d4)

Installing 🚀

```
git clone --no-single-branch --depth 50 https://github.com/Jakob933/ccsdk-cds.git .  
git clone --no-single-branch --depth 50 https://github.com/Jakob933/ccsdk-cds.git .  
git clone --no-single-branch --depth 50 https://github.com/Jakob933/ccsdk-cds.git .  
git clone --no-single-branch --depth 50 https://github.com/Jakob933/ccsdk-cds.git .  
git checkout --force origin/master  
git clean -d -f -f  
python3.7 -mvirtualenv /home/docs/checkouts/readthedocs.org/user_builds/onap-cds/envs/latest  
/home/docs/checkouts/readthedocs.org/user_builds/onap-cds/envs/latest/bin/python -m pip install --
```

Links

- Read The Docs / Confluence Policy:
<https://wiki.onap.org/display/DW/DeveloperWiki+and+ReadTheDocs+Usage+Policy>
- Migration of Developer Wiki Content to Read The Docs:
<https://wiki.onap.org/display/DW/Migration+of+DeveloperWiki+Content+to+ReadTheDocs>
- Creating Documentation in Read The Docs:
<https://docs.onap.org/en/latest/guides/onap-developer/how-to-use-docs/index.html>
- Confluence pages to .rst converter:
<https://wiki.onap.org/display/DW/Doc+Tools>
- Tabbed view in RDT:
<https://github.com/executablebooks/sphinx-tabs>
- VSC reStructuredText Extension:
<https://marketplace.visualstudio.com/items?itemName=lexstudio.restructuredtext>

Thank you!