**WWN Development Document**

The WWN input parameter-file:

YAML-tool use, and YAML-tool references

Word’s Navigation pane shows the table-of-contents (View : Show : Navigation pane).

* Contents:
  + For the WWN input parameter-file, describes:
    - How the YAML tools are used to verify the contents, and load them into a Python dictionary
  + References for the YAML tools, e.g., user manuals and tutorials

This document was created by the WWN author for his own use in developing WWN. It is included in the WWN repo, as other developers may find it useful.

# The WWN input parameter-file: YAML-tool use

WWN uses an input parameter-file, and it is in YAML format. This section describes how the input parameter-file is processed, and how the YAML tools are used. The input parameter-file is loaded and verified in load\_input\_parameter\_file.py.

When create\_web\_page is called, a parameter-file is specified. It specifies the Word HTML file to be processed, and how it’s to be processed. This parameter-file is referred to as the *input* *parameter-file*, or simply, *the parameter-file*.

The parameter-file is in YAML format. A Python library is used to load the YAML file, and two libraries are used to verify the parameter-file's syntax.

* *yamllint* verifies that proper YAML syntax is used.
* *PyYaml* is used to load the YAML file's contents into a Python dictionary
* *Cerberus* verifies the YAML keys and values that are defined for create\_web\_page, e.g., Cerberus verifies required keys are specified.

In the YAML file-format, data is structured as dictionaries and lists. In Python, when the YAML file is loaded, it is stored as dictionaries and lists.

yamllint itself uses a config-file, and that config-file specifies what syntax checking yamllint should perform. A yamllint config-file is distributed with create\_web\_page, which it uses. By default, yamllint's syntax requirements are excessive, and that config-file is primarily used to relax those requirements. If yamllint finds a syntax error, the error is reported on the console, and create\_web\_page exits. The error message returned by yamllint is displayed.

PyYaml has a function yaml.load(), and it is used to load the parameter-file. yaml.load() is called after calling yamllint. yaml.load() also performs YAML syntax checking. If yaml.load() finds a syntax error, the error is reported on the console, and create\_web\_page exits. The error message returned by yaml.load() is displayed.

Cerberus is called to verify the parameter-file, via a schema. The schema is specified in load\_input\_parameter\_file.py. The schema specifies the keys and key-values for the parameter-file. This includes the keys' names and the structure of the keys (e.g., for a particular key, what keys are under it). The schema also specifies whether a key is required, and the syntax for the key's value (e.g., a string or number). If Cerberus finds a syntax error, the error is reported on the console, and create\_web\_page exits. The error message returned by Cerberus is displayed.

The Cerberus error-messages are formatted using pprint++, a pretty-printer app.

# References

* YAML syntax
  + YAML Syntax — Ansible Documentation
    - <https://docs.ansible.com/ansible/latest/reference_appendices/YAMLSyntax.html>
  + YAML Tutorial - Tutorialspoint
    - <https://www.tutorialspoint.com/yaml/index.htm>
* yamllint references:
  + Documentation: <https://yamllint.readthedocs.io/en/stable/>
  + Calling yamllint from Python:
    - <https://yamllint.readthedocs.io/en/stable/development.html>
    - <https://github.com/adrienverge/yamllint/issues/112>
  + Configuration: <https://yamllint.readthedocs.io/en/stable/configuration.html>
    - Default configuration:
      * <https://yamllint.readthedocs.io/en/stable/configuration.html#default-configuration>
      * <https://github.com/adrienverge/yamllint/blob/master/yamllint/conf/default.yaml>
    - Extending the default configuration:
      * <https://yamllint.readthedocs.io/en/stable/configuration.html#extending-the-default-configuration>
    - YamlLintConfig(): <https://github.com/adrienverge/yamllint/blob/master/yamllint/config.py>
  + Verification rules: <https://yamllint.readthedocs.io/en/stable/rules.html>
* PyYaml references:
  + Documentation: <https://pyyaml.org/wiki/PyYAMLDocumentation>
  + yaml.load() errors are returned via exceptions. This includes YAML syntax errors that are detected.
    - <https://pyyaml.org/wiki/PyYAMLDocumentation>
    - <https://stackoverflow.com/questions/30269723/how-to-get-details-from-pyyaml-exception>
* Cerberus references:
  + Documentation: <https://docs.python-cerberus.org/en/stable/api.html>
  + Validation Rules: <https://docs.python-cerberus.org/en/latest/validation-rules.html#of-rules>
    - This web-page defines the schema functions that can be used to enforce syntax rules.
  + How-to define multipe schemas for a single field (e.g., a list whose members are of different types)
    - <https://stackoverflow.com/questions/50499045/python-cerberus-multipe-schemas-for-a-single-filed/50506324#50506324>
* pprint++ references
  + <https://github.com/wolever/pprintpp>