

The following guide provides patterns for interface elements and directories.

Operator Menus

Each operator should receive its own heading or page based on the length of the content. At the start should be a reference admonition documenting the context of the operator:

```
.. reference::

:Mode:      Edit Mode
:Menu:      :menuselection:`Curve --> Snap`
:Shortcut:  :kbd:`Shift-S`
```

Panels

Panels should be documented by their own heading, nested panels should use decreasing heading levels. Each panel could have its own page based on the length of documentation and/or the amount of panels. Expanded menus that toggle what properties are presented to the user should be treated like subpanels:

```
Panel Title
=====

Nested Panel Title
-----
```

Properties

Properties should be documented using definition lists. Properties that are hidden based on other properties should use nested definitions:

```
Property
  Property description.

Hidden Property
  Hidden property description.
```

Select menus should be documented using the following syntax:

```
Menu Label
  General description of the menu.

:Menu Item: Menu Item Definition.
:Menu Item: Menu Item Definition.
:Menu Item: Menu Item Definition.
```

Nodes

Nodes should always have three headings inputs, properties and outputs with a note of absence if the node has none. At the end of the page can be an optional example(s) section:

```
*****
Example(s)
```

```

World Node
*****

.. figure:: /images/render_shader-nodes_output_world_node.png
   :align: right

   The World node.

Introduction and general use case(s).


Inputs
=====

This node has no inputs.


Properties
=====

This node has no properties.


Outputs
=====

This node has no outputs.


Example
=====

```

Directory Layout

Sections should be generally structured as follows:

- directory_name/
 - index.rst (contains links to internal files)
 - introduction.rst
 - section_1.rst
 - section_2.rst

For example:

- rendering/
 - index.rst
 - cycles/
 - index.rst
 - introduction.rst
 - materials/
 - index.rst
 - introduction.rst
 - volumes.rst

The idea is to enclose all the content of a section inside of a folder. Ideally every section should have an `index.rst` (containing the TOC for that

section) and an `introduction.rst` (introducing) to the contents of the section.

Table of Contents

By default, a table of contents should show two levels of depth:

```
.. toctree::
   :maxdepth: 2

   introduction.rst
   perspective.rst
   depth_of_field.rst
```