



From Manual Testing to Continuous Validation: Taking the Quality of Ansible Content to the Next Level

Sorin Sbarnea

Principal Software Engineer // Ansible DevTools Team Lead

GitHub: [ssbarnea](#)

Matrix: [@ssbarnea:matrix.org](#)



I know how to create the content, but...

- How do I maintain it easily?
- How to minimize my effort?
- How do I know if a change will break it or not?
- How do I make it future proof?



Deal with the cost of fame...

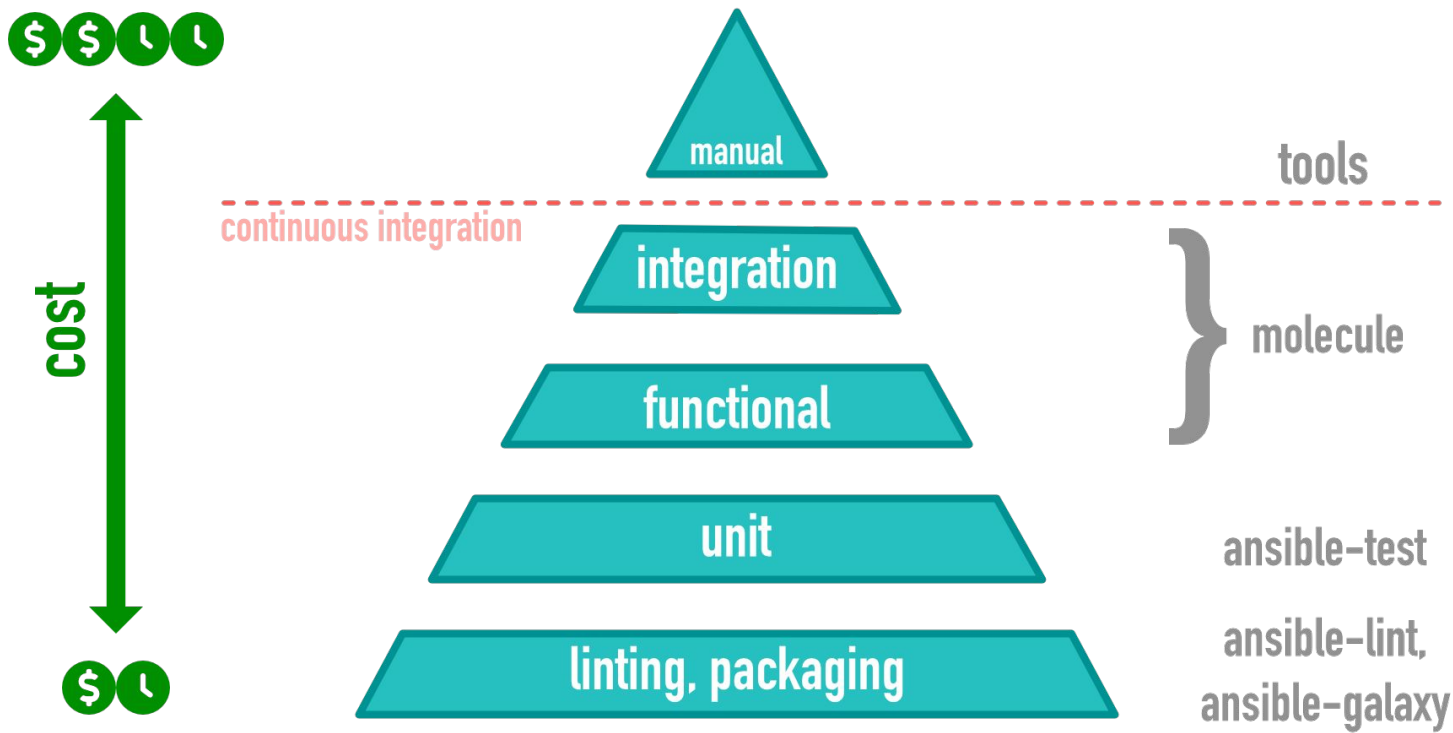
- More users are consuming your content
- Some are using in genuine unsupported ways
- More bugs are reported
- More feature requests are posted
- ... this becomes your 2nd job!



Use CI/CD to test any change

- Do not trust someone saying "works on my machine"
- Trust CI/CD jobs... but watch them closely
- Test against **current** and **upcoming** versions

Testing pyramid





Start from the bottom

- **Static analysers and formatters**
 - ansible-lint
 - ansible test sanity
 - yamllint
 - black
 - other pre-commit based linters
- **Unit testing**
 - pytest-ansible
 - ansible-test unit
- **Functional and integration testing**
 - molecule
 - ansible-test integration




First CI/CD pipeline to implement

- Ansible-lint is a meta-linter
 - Covers most static analysis needed for Ansible content
- It is very easy to integrate
- Easy to maintain
- Low testing resource use
- Action is releases at the same time as lint itself
- You can decide to use main branch or a stable release

<https://github.com/marketplace/actions/run-ansible-lint>

Minimal workflow

```
# .github/workflows/ansible-lint.yml
name: ansible-lint
on:
  pull_request:
jobs:
  build:
    name: Ansible Lint
    runs-on: ubuntu-24.04
    steps:
      - uses: actions/checkout@v4
      - name: Run ansible-lint
        uses: ansible/ansible-lint@main # or @v25 or @v25.1.1
```



26%
@main



Set of github actions used by networking team

- Allow use of ansible-lint
- Allow running ansible-test sanity, unit and functional
- Used in production by >10 important collections



Leverage use of tox-ansible

- tox-ansible is not only for local testing
- It can also generate github test matrix jobs!
- You no longer need to update GitHub workflows for:
 - ansible versions
 - python versions
- You control the test matrix locally



```
$> tox list
```

```
default environments:
```

```
integration-py3.13-2.18 -> Integration tests using ansible-core 2.18 and python 3.13
integration-py3.13-devel -> Integration tests using ansible-core devel and python 3.13
sanity-py3.13-2.18      -> Sanity tests using ansible-core 2.18 and python 3.13
sanity-py3.13-devel    -> Sanity tests using ansible-core devel and python 3.13
unit-py3.13-2.18       -> Unit tests using ansible-core 2.18 and python 3.13
unit-py3.13-devel      -> Unit tests using ansible-core devel and python 3.13
```

```
# ^ normal test matrix much bigger but it can be reduced using the "skip" option in config
```



```
[
  {
    "description": "Integration tests using ansible-core 2.18 and python 3.13",
    "factors": [
      "integration",
      "py3.13",
      "2.18"
    ],
    "name": "integration-py3.13-2.18",
    "python": "3.13"
  },
  {
    "description": "Integration tests using ansible-core devel and python 3.13",
    "factors": [
      "integration",
      "py3.13",
      "devel"
    ],
    "name": "integration-py3.13-devel",
    "python": "3.13"
  },
  ...
]
```



v2 will mean simplification

- If you already use ansible-content-actions
- Add or removing an ansible version will be done inside tox configuration
- No need to touch the workflow file
- Our ansible-creator templates will include these workflows
- Watch the repository to get notified when we release the new actions



- Ability to chain multiple tests on the same github runner
- Reduce number of jobs from >40 to $\frac{1}{3}$
- Allow uses to decide if they want to make use of chaining



Focus areas in the year ahead:

- Creating automation content
- Testing, releasing, and publishing collections
- Providing consistent developer experiences

Let us know what you think:

- What enhancements or new features do you want to see?
- What do you think works well currently?
- What are your biggest pain points?

Forum post: [Ansible DevTools plans and collaboration opportunities](#)





Anyone can contribute.
We are all community.

Join the forum to participate
in discussions and get help!



Want to contribute? Find out
how to get involved.





Thanks!

GitHub: **ssbarnea**

Matrix: **#community:ansible.com #social:ansible.com**

Ansible community forum:
<https://forum.ansible.com/>

ANSIBLE