



Ansible Core Architecture: Handlers

Martin Krizek

Principal Software Engineer, Ansible Core

GitHub: [mkrizek](#)



- Understanding Handlers
- Handlers Architecture Then and Now



Understanding Handlers

The Basics: Defining and Notifying Handlers



```
- hosts: all
  tasks:
    - dnf:
      name: cowsay
      notify:
        - by_name
        - topic
  handlers:
    - name: by_name
      debug:
        msg: cannot be notified, the next one overrides it
    - name: by_name
      debug:
        msg: can be notified by its name
    - name: by_name_and_topic
      debug:
        msg: can be notified by both name and topics
      listen:
        - topic
        - topic2
    - debug:
      msg: can be notified by its topic
      listen: topic
```

```
TASK [dnf] *****
changed: [localhost]
```

```
RUNNING HANDLER [by_name]
*****
ok: [localhost] => {
  "msg": "can be notified by its name"
}
```

```
RUNNING HANDLER [by_name_and_topic]
*****
ok: [localhost] => {
  "msg": "can be notified by both name and topics"
}
```

```
RUNNING HANDLER [debug]
*****
ok: [localhost] => {
  "msg": "can be notified by its topic"
}
```

1
2
3
4



```
- hosts: localhost
  force_handlers: false # or --force-handlers on the cmd line or force_handlers in [defaults] in ansible.cfg
  pre_tasks:
    . . . list of tasks
    - name: explicitly run handlers notified up to this point
      meta: flush_handlers
    . . . list of tasks
    - meta: flush_handlers # added implicitly by ansible-core
  roles:
    . . . list of roles
  tasks:
    . . . list of tasks
    - meta: flush_handlers # added implicitly by ansible-core
  post_tasks:
    . . . list of tasks
    - meta: flush_handlers # added implicitly by ansible-core
```



```
% cat playbook.yml
```

```
- hosts: localhost
  roles:
    - role_a
  tasks:
    - command: echo
      notify:
        - handler1
        - "role_a : handler1"
  handlers:
    - name: handler1
      debug:
        msg: handler1 from the handlers section
```

```
% cat roles/role_a/handlers/main.yml
```

```
- name: handler1
  debug:
    msg: handler1 from a role
```

Order of Definition == Order of Execution



```
- hosts: localhost
roles:
  - r_roles
tasks:
  - include_role:
      name: r_include_role

  - import_role:
      name: r_import_role

- command: echo
  notify:
    - handler_from_import_role
    - handler_from_handlers
    - handler_from_include_role
    - handler_from_roles

handlers:
  - name: handler_from_handlers_section
    debug:
      msg: handler_from_handlers_section
```

```
RUNNING HANDLER [r_roles : handler_from_roles]
*****
ok: [localhost] => {
    "msg": "handler_from_roles"
}
```

```
RUNNING HANDLER [handler_from_handlers_section]
*****
ok: [localhost] => {
    "msg": "handler_from_handlers_section"
}
```

```
RUNNING HANDLER [r_import_role : handler_from_import_role]
*****
ok: [localhost] => {
    "msg": "handler_from_import_role"
}
```

```
RUNNING HANDLER [r_include_role : handler_from_include_role]
*****
ok: [localhost] => {
    "msg": "handler_from_include_role"
}
```



```
- hosts: localhost
  tasks:
    - dnf:
      name: cowsay
      state: present
    notify:
      - include_tasks_as_handler      # works
      - import_tasks_as_handler      # no such handler
      - task_from_within_dynamic_yaml # no such handler
      - task_from_within_static_yaml # works

  handlers:
    - name: include_tasks_as_handler
      include_tasks:
        name: dynamic.yml

    - name: import_tasks_as_handler
      import_tasks:
        name: static.yml
```

“include_tasks is a handler, import_tasks creates handlers”

Handlers Notifying Handlers



```
- hosts: localhost
tasks:
  - name: tasks_command
    command: echo
    notify: handler2
  - meta: flush_handlers # added implicitly by ansible-core

post_tasks:
  - name: post_tasks_command
    command: echo
  - meta: flush_handlers # added implicitly by ansible-core

handlers:
  - name: handler1
    debug:
      msg: handler1
  - name: handler2
    command: echo
    notify:
      - handler1 # due to flush_handlers in post_tasks
      - handler3
  - name: handler3
    debug:
      msg: handler3
```

```
TASK [tasks_command]
*****
changed: [localhost]

RUNNING HANDLER [handler2]
*****
changed: [localhost]

RUNNING HANDLER [handler3]
*****
ok: [localhost] => {
  "msg": "handler3"
}

TASK [posts_task_command]
*****
changed: [localhost]

RUNNING HANDLER [handler1]
*****
ok: [localhost] => {
  "msg": "handler1"
}
```



- `include_role` **and** `import_role` **as handlers**
- `meta: flush_handlers` **as a handler**
- **Tags**
- `block:`, `rescue:` **and** `always:` **sections**
- **No templates in** `listen`
- **Beware of templates in handler names**

Ref: https://docs.ansible.com/ansible/latest/playbook_guide/playbooks_handlers.html



Handlers Architecture Then and Now




- Ignores the specified execution strategy plugin (`free`, `host_pinned`, ...)
- Some task keywords unavailable (`run_once`, `any_errors_fatal`, `serial`, ...)
- Cannot use meta tasks as handlers
- Tags
- Single `flush_handlers` execution for all hosts, no `when conditional`


```
def flush_handlers():
    # FIXME this should not be its own strategy
    for handler in Play.handlers:
        for host in handler.notified_hosts:
            send_task_to_worker(handler, host)

    # FIXME this code is already in each strategy
    process_and_run_handlers_from_include_tasks()
```



Move handler processing into new PlayIterator state

 Merged mkrizek merged 44 commits into `ansible:devel` from `mkrizek:iterating_handlers_light`

 Conversation 45

 Commits 44

 Checks 0

 Files changed 28



mkrizek commented on Jun 2, 2022 · edited ▾

SUMMARY

Fixes [#46447](#)

Fixes [#52561](#)

Fixes [#54991](#)

Fixes [#64611](#)

Fixes [#64999](#)

Fixes [#65067](#)

Fixes [#72725](#)

Fixes [#72781](#)

Fixes [#77616](#)



- Now regular tasks going through strategy plugins
- `PlayIterator`
- Functionality for free
 - task keywords
 - meta tasks as handlers
- Changes in behavior aligned with the documentation
 - Mostly around “the last one defined with the same name wins” for roles and `listen`
 - https://docs.ansible.com/ansible/latest/porting_guides/core_porting_guides.html

```
def flush_handlers(host):  
    if evaluate(when):  
        PlayIterator.set_state_for_host(host, state=HANDLERS)
```



- `meta: flush_handlers` optimizations
- `listen` validation at parse time



Thanks!

GitHub: **mkrizek**

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

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

ANSIBLE