

# GlobalNOC Network Automation Tools

AJ Ragusa

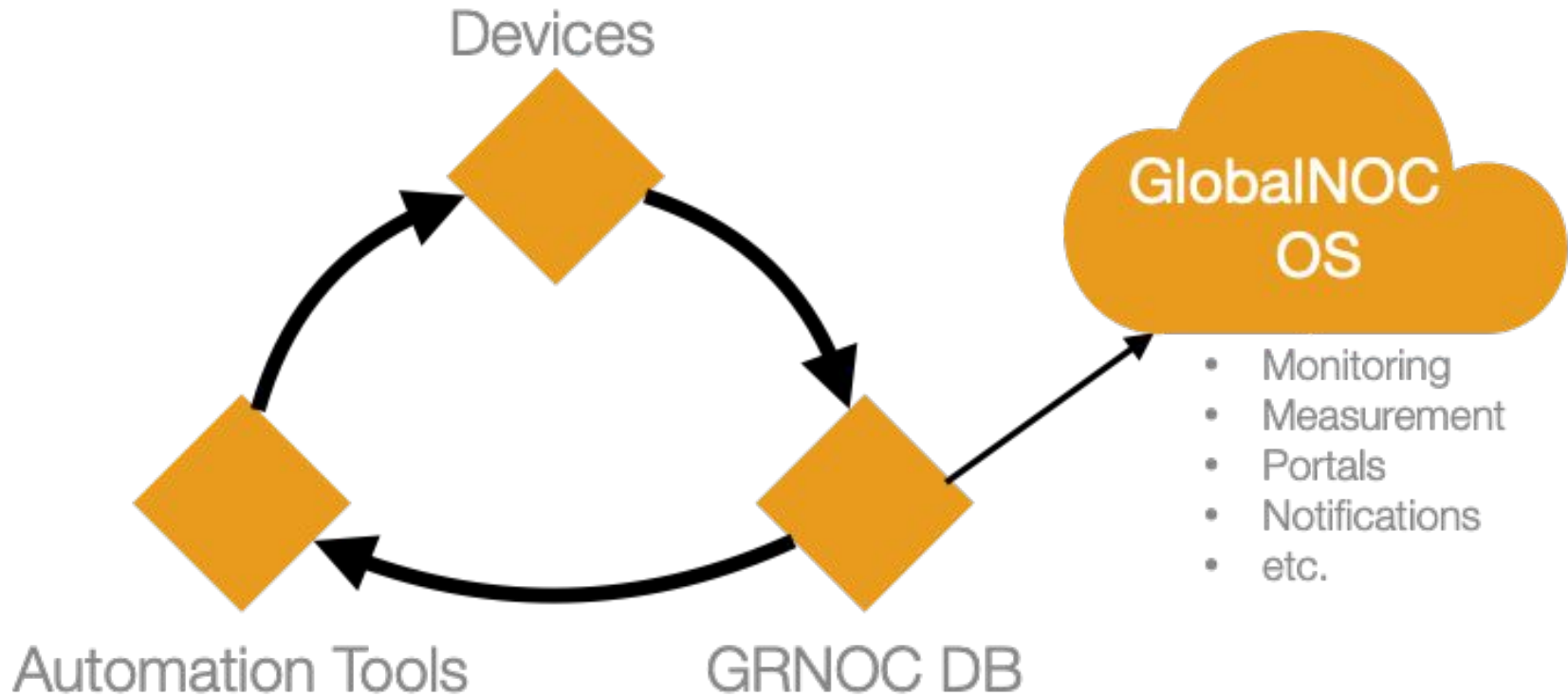
# Philosophy / Goals

---

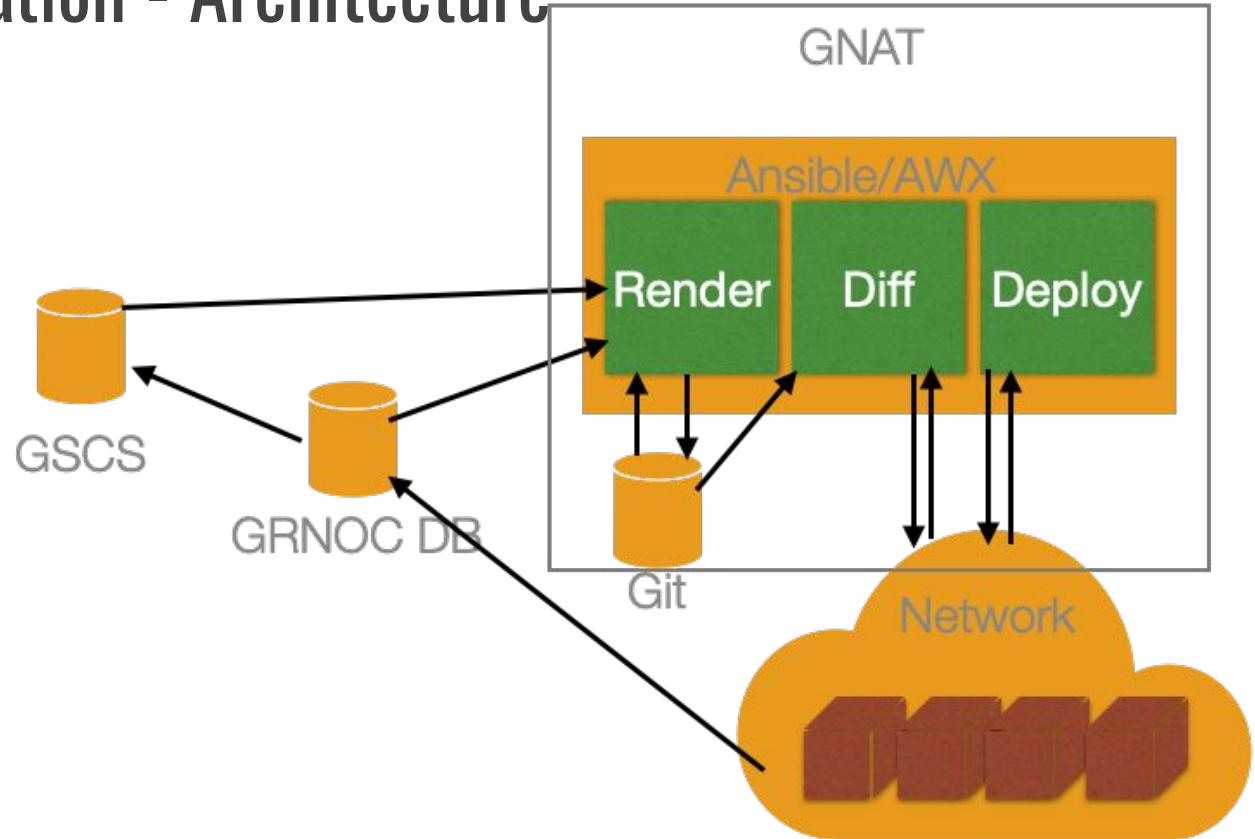
- Flexibility
  - Automate as much or as little of your network as you want
  - Ability to modify the workflow (add in pre/post checks, reviewers, etc.)
  - Software changes not required for new services/models
- Fix Security, and reduce troubleshooting times through standardization
- Treat diverse parts of the configuration as a single “block” of config
- Model/Store this data in a vendor agnostic format, but show vendor specific
- Speed up deployment of new services
- Get the Network Engineers to WANT to use the automation tools!

# Automation Lifecycle

---



# Network Automation - Architecture



# GNAT - GlobalNOC Network Automation Tool

- A combination of “static” templates filled with information from GRNOC DB
- Ansible playbooks / roles are included in the repository
- AWX / Ansible tower is how we launch our playbooks (backend)
- Custom Web-UI facilitates workflow with Network Engineers
- Network Engineers now have full ability to edit the Templates/Plays
- Can integrate other sources (like GSCS, or RADB)
- Add reviewers and add pre-post checks or other checkpoints into workflow

# GNAT - Many different tools - 1 workflow

---

- AWX/Ansible - provides the automation engine for GNAT
  - AWX provides a centralized location for adding new playbooks, storing credentials for logging in, and launching jobs via Web-Services
- Git/GitHub - provides a central repository for storing device configuration and playbooks
  - Stores templates, configurations, passwords and playbooks
  - Versioned! We can roll back to a different version of the network
- GNAT WebUI - The place to launch the workflows

# Jinja - YAML - JSON training

---

See Other slide deck!

# Git/GitHub - Version control for Network Engineers

---

See Slide deck!



# AWX/Ansible

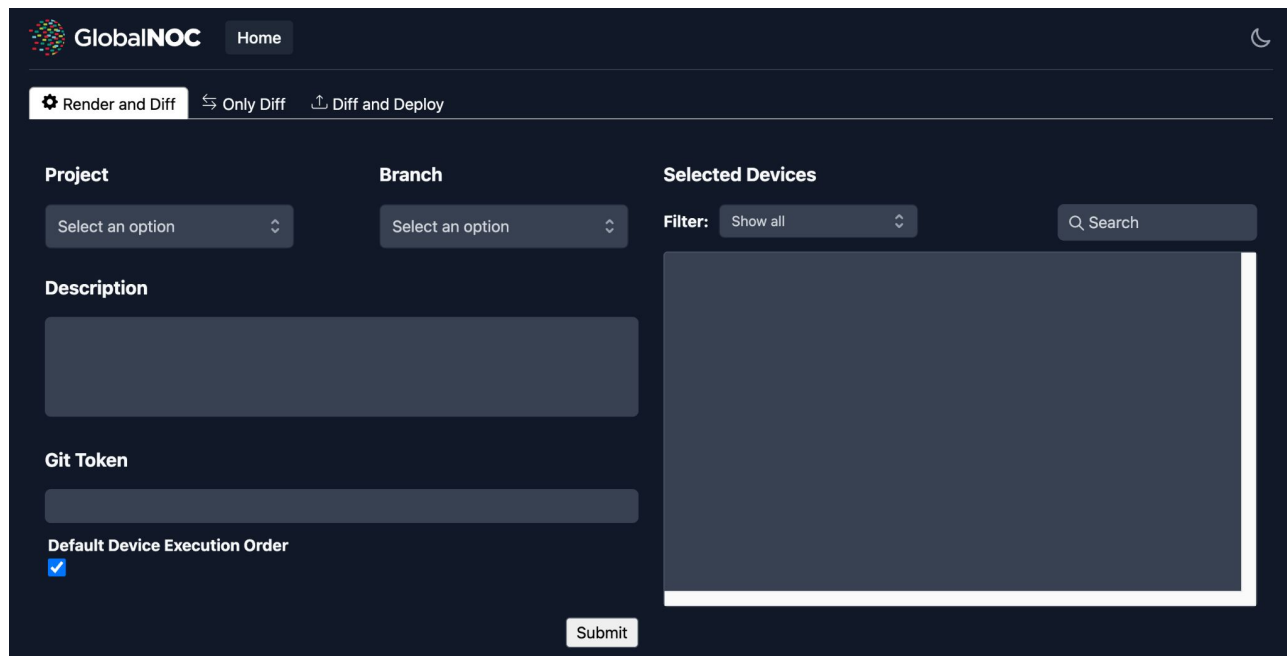
---

See more slides!

# Finally GNAT!

---

- Git Repository Structure
- Templates
- Variables
- The Jobs
  - Render + Diff
  - Diff + Deploy



The screenshot displays the GlobalNOC web interface. At the top left is the GlobalNOC logo and a 'Home' button. A navigation bar contains three buttons: 'Render and Diff' (active), 'Only Diff', and 'Diff and Deploy'. The main form area is divided into three columns: 'Project', 'Branch', and 'Selected Devices'. Each column has a dropdown menu labeled 'Select an option'. The 'Selected Devices' column includes a 'Filter' dropdown set to 'Show all' and a search input field. Below these columns are three text input fields labeled 'Description', 'Git Token', and 'Default Device Execution Order' (which has a checked checkbox). A 'Submit' button is located at the bottom right of the form.

# What is GNAT's Role?

---

- All of the things that are “the same” across a node role
  - Eg. RADIUS config, NTP, Syslog
  - iBGP, root password, prefix-lists, ACLs, etc...
- Most Networks call this the “base-line”
- What GNAT is not for
  - Each interface's description
  - Individual BGP peers
  - Services that aren't on every node! (this is what GSCS is for)

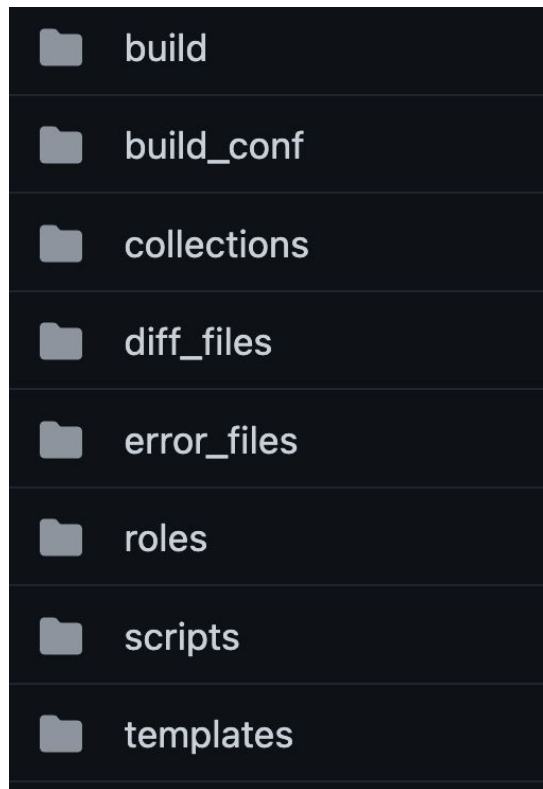
# Common Use Cases

---

- I need to update the root password on all the devices
- I need to add an allowed SSH login host to all devices
- The address of the syslog server has changed
- Diff expected configuration to what is deployed

All of these use cases are great for this tool. Even more we can verify the configuration deployed on a device MATCHES what we expect to be there (raise an alarm in Alertmon if it doesn't match)

# GitHub Repository Structure - GNAT



build
build_conf
collections
diff_files
error_files
roles
scripts
templates

The branches rendered configuration - each device will have its own .conf file

Configuration variables used to build the configurations (passwords, globals, etc.)

For Ansible - you can ignore

Location of Diff files (when diff runs)

For Ansible - you can ignore

For Ansible - you can ignore

Scripts maintained by syseng (you can ignore)

Where templates are stored! - YOU WANT THIS!

# Templates! - Finally we have enough to work with these!

Please note the path!

Each role code has its own directory

The screenshot shows the GitHub interface for the repository `OSHEAN/OSHEAN-Network`. The breadcrumb path is `OSHEAN-Network / templates / role /`. Below the breadcrumb, a commit by `gcbrowni` is shown with the message `Create ncs-mp.j2`. The commit details table lists the following files:

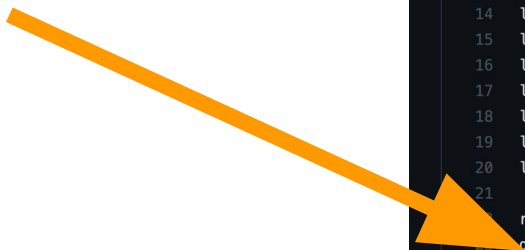
File	Commit Message	Time
<code>ncs-core</code>	Create <code>ncs-core.j2</code>	17 months ago
<code>ncs-mp</code>	Create <code>ncs-mp.j2</code>	17 months ago
<code>rtsw</code>	Adding initial setup	2 years ago
<code>testlab</code>	Update <code>testlab.j2</code>	2 years ago

# Lets do a quick example of how we would change something!

---

Specifically lets change the name-servers for the ncs-core devices!

Change this!



```
newCore-and-mp ▾ OSHEAN-Network / templates / role / ncs-core / ncs-core.j2
gcbrowni Create ncs-core.j2
1 contributor
240 lines (189 sloc) 5.45 KB
1  replace: hostname
2  hostname {{ inventory_hostname | regex_replace('^(ncs-core[0-9]?[0-9]*)\.mgmt\.oshean\.org$', '\\1') }}
3
4  replace: clock
5  clock timezone EST America/New_York
6
7  replace: service
8  service timestamps log datetime localtime msec
9  service timestamps debug datetime localtime msec
10
11 replace: logging
12 logging trap critical
13 logging console debugging
14 logging monitor debugging
15 logging 140.182.49.54 vrf MgmtNet severity info
16 logging 131.109.204.45 vrf MgmtNet severity info
17 logging 131.109.204.54 vrf MgmtNet
18 logging 158.123.143.250 vrf default severity info
19 logging source-interface Loopback1 vrf MgmtNet
20 logging events link-status software-interfaces
21
22 replace: domain
23 domain vrf MgmtNet name mgmt.oshean.org
24 domain vrf MgmtNet name-server 8.8.8.8
```

# Closer look

Hey What is this! Its not valid config!

```
-----  
22  replace: domain  
23  domain vrf MgmtNet name mgmt.oshean.org  
24  domain vrf MgmtNet name-server 8.8.8.8  
25  domain vrf MgmtNet name-server 8.8.7.7  
26  domain vrf MgmtNet lookup source-interface Loopback1  
27  
28  replace: telnet  
29  telnet vrf default ipv4 server max-servers 100
```



# Replace statement

---

- In IOS-XR and JunOS devices the replace statement operates similarly.
  - In Junos it will replace everything underneath that structure with whatever is specified there
  - In IOS-XR it essentially will append a 'no' to every line of the current config that matches (ie.. 'replace domain' will read the current configuration of the device and find all statements that match 'domain\*' and prepend them with a 'no' before the rest of your config goes in

## We can edit the config

---

```
22  replace: domain
23  domain vrf MgmtNet name mgmt.oshean.org
24  domain vrf MgmtNet name-server 1.1.1.1
25  domain vrf MgmtNet name-server 8.8.7.7
26  domain vrf MgmtNet lookup source-interface Loopback1
27
```

# Now with our change we go to the GNAT UI

The screenshot displays the GlobalNOC GNAT UI interface. At the top left is the GlobalNOC logo and a 'Home' button. Below the navigation bar are three buttons: 'Render and Diff' (with a gear icon), 'Only Diff' (with a double arrow icon), and 'Diff and Deploy' (with an up arrow icon). The main content area is divided into three columns: 'Project', 'Branch', and 'Selected Devices'. The 'Project' dropdown is set to 'OSHEAN'. The 'Branch' dropdown is set to 'newCore-and-mp'. The 'Selected Devices' section has a 'Filter' dropdown set to 'Show all' and a search bar. A list of devices is shown, all with checked selection boxes and 'active' status. The 'Description' field contains the text 'This is an example!' and a green 'G' icon with a '1' notification bubble. The 'Git Token' field is empty. The 'Default Device Execution Order' checkbox is checked. A 'Submit' button is located at the bottom center.

**GlobalNOC** Home

Render and Diff Only Diff Diff and Deploy

**Project** OSHEAN

**Branch** newCore-and-mp

**Selected Devices** Filter: Show all Search

**Description**  
This is an example!

**Git Token**  
.....

**Default Device Execution Order**

**Selected Devices List:**

- ncs-core
  - ncs-core1.nav400min.mgmt.oshean.org active
  - ncs-core1.ner1summer.mgmt.oshean.org active
  - ncs-core1.osh210benef.mgmt.oshean.org active
  - ncs-core1.osh235prome.mgmt.oshean.org active
  - ncs-core1.sto320washn.mgmt.oshean.org active
  - ncs-core1.uri1bairdhi.mgmt.oshean.org active
  - ncs-core1.whe26emain.mgmt.oshean.org active
  - ncs-core2.nav400min.mgmt.oshean.org active
  - ncs-core2.ner1summer.mgmt.oshean.org active
  - ncs-core2.osh210benef.mgmt.oshean.org active
  - ncs-core2.osh235prome.mgmt.oshean.org active
  - ncs-core2.sto320washn.mgmt.oshean.org active

Submit

Select Render and Diff

Select the Project and Branch

Select Devices to run it on!

The screenshot shows the GlobalNOC interface with the following components:

- GlobalNOC Header:** Includes the logo, a 'Home' button, and a moon icon for dark mode.
- Navigation Bar:** Contains a gear icon for 'Render and Diff', a left arrow for 'Only Diff', and an up arrow for 'Diff and Deploy'.
- Project Selection:** A dropdown menu currently showing 'OSHEAN'.
- Branch Selection:** A dropdown menu currently showing 'newCore-and-mp'.
- Selected Devices Section:**
  - Filter:** A dropdown menu set to 'Show all'.
  - Search:** A search input field with a magnifying glass icon.
  - Device List:** A scrollable list of devices, each with a checked checkbox and a status label 'active'. The list includes:
    - ncs-core
    - ncs-core1.nav400min.mgmt.oshean.org
    - ncs-core1.ner1summer.mgmt.oshean.org
    - ncs-core1.osh210benef.mgmt.oshean.org
    - ncs-core1.osh235prome.mgmt.oshean.org
    - ncs-core1.sto320washn.mgmt.oshean.org
    - ncs-core1.uri1bairdhi.mgmt.oshean.org
    - ncs-core1.whe26emain.mgmt.oshean.org
    - ncs-core2.nav400min.mgmt.oshean.org
    - ncs-core2.ner1summer.mgmt.oshean.org
    - ncs-core2.osh210benef.mgmt.oshean.org
    - ncs-core2.osh235prome.mgmt.oshean.org
- Description:** A text area containing 'This is an example!' and a green 'G' icon with a '1' notification badge.
- Git Token:** A text input field with a masked password '.....'.
- Default Device Execution Order:** A checkbox that is checked.
- Submit:** A button at the bottom center.

# While it is running

Shows its running, complete or failed

Real-time job status

Resulting diffs will appear here!

The screenshot displays the GlobalNOC interface. At the top, there is a navigation bar with the GlobalNOC logo and a 'Home' button. Below the navigation bar, there are two main panels. The left panel shows a job titled 'newnewCore-and-mp / OSHEAN Render (running)' with a 'Cancel' button. The job output includes a warning about invalid characters in group names and two lines of output: 'PLAY [Stuff to run once on the localhost] \*\*\*\*\*' and 'TASK [Template global variables] \*\*\*\*\*'. Below the job output, there is a section for 'Diff playbook' with a search bar. The right panel is titled 'Diff Files' and contains a search bar labeled 'Search diffs'. Three orange arrows point from the text annotations to the job title, the job output, and the 'Search diffs' bar.

GlobalNOC Home

newnewCore-and-mp / OSHEAN Render (running) Cancel

Vault password:  
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details

PLAY [Stuff to run once on the localhost] \*\*\*\*\*

TASK [Template global variables] \*\*\*\*\*

Diff playbook

Diff Files

Search diffs

# Success running Render - now doing diff

The screenshot shows the GlobalNOC interface. At the top left is the GlobalNOC logo and a 'Home' button. At the top right is a moon icon. The main content area is divided into two panels. The left panel shows a list of tasks: a successful 'newnewCore-and-mp / OSHEAN Render' and a running 'newnewCore-and-mp / OSHEAN Diff'. A 'Cancel' button is next to the diff task. Below the tasks is a terminal window showing the output of the diff process. The right panel is titled 'Diff Files' and contains a search bar for diff files.

GlobalNOC Home

> ✓ newnewCore-and-mp / OSHEAN Render (successful)

∨ 🔄 newnewCore-and-mp / OSHEAN Diff (running) Cancel

```
SSH password:
[WARNING]: Invalid characters were found in group names but not replaced, use
-vvvv to see details

PLAY [localhost] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Delete old diff file if it exists] *****
changed: [localhost]

TASK [create diff directory] *****
changed: [localhost]

PLAY [Stuff to run on the localhost once per host] *****
```

Diff Files

🔍 Search diffs

# Finally Results!

— — —



Home



- newnewCore-and-mp / OSHEAN Render (successful)
- newnewCore-and-mp / OSHEAN Diff (successful)

## Diff Files

🔍 Search diffs

### ncs-core.ccri1jchafe.mgmt.oshean.org.diff

[View on GitHub](#)

```
Diff Timestamp: Thu Nov 3 02:58:13 UTC 2022
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:2a88c9c8-c84e-
49eb-a705-d8572a2a6739">
  <response xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-cli-diff-act">
    Building configuration...
    !! IOS XR Configuration 7.4.2
    + service unsupported-transceiver
    + domain name mgmt.oshean.org
    + domain name-server 158.123.143.171
    + domain name-server 158.123.143.172
    + domain lookup disable
    line default
    + login authentication default
    !
    &lt;- snmp-server location CCRI_Newport
    +&gt; snmp-server location ncs-core.ccri1jchafe
    ntp
```

# Lets look at Diff and Deploy

Only devices currently rendered on this branch can be deployed

The screenshot shows the GlobalNOC interface for the 'Diff and Deploy' action. At the top, there are three tabs: 'Render and Diff', 'Only Diff', and 'Diff and Deploy'. Below the tabs, the 'Project' is set to 'OSHEAN' and the 'Branch' is 'newnewCore-and-mp'. There are links for 'Manually Resolve Conflicts' and 'Auto-Resolve Conflicts'. The 'Description' field is empty. The 'Git Token' field is also empty. The 'Deployment Batch Size' is set to 1, and the 'Rollback Time (in minutes)' is set to 30. There are checkboxes for 'Dry Run' (unchecked) and 'Pause Between Batches' (checked). The 'Device Execution Order' section is partially visible. On the right, the 'Selected Devices' list is shown with a 'Filter' dropdown set to 'Show all' and a search bar. The list contains various device names with checkboxes and status indicators (active or install).

How long to wait for a confirmation or rollback

Batch size (how to break apart the

Device Name	Status
ncs-core2.ner1summer.mgmt.oshean.org	active
ncs-core2.osh210benef.mgmt.oshean.org	active
ncs-core2.osh235prom.mgmt.oshean.org	active
ncs-core2.sto320washn.mgmt.oshean.org	install
ncs-core3.osh210benef.mgmt.oshean.org	install
ncs-core3.osh235prom.mgmt.oshean.org	active
ncs-core4.osh235prom.mgmt.oshean.org	active
ncs-core5.osh235prom.mgmt.oshean.org	active
ncs-core6.osh235prom.mgmt.oshean.org	active
ncs-core7.osh235prom.mgmt.oshean.org	install
ncs-core.ccri1jchafe.mgmt.oshean.org	active
ncs-core.ccri400east.mgmt.oshean.org	active
ncs-core.rwu1oldfer.mgmt.oshean.org	active
ncs-core.uri109tyler.mgmt.oshean.org	active
ncs-mp	
ncs-mp1.aos1dorranc.mgmt.oshean.org	active
ncs-mp1.aos222quakr.mgmt.oshean.org	active
ncs-mp1.aos24weybos.mgmt.oshean.org	active
ncs-mp1.aos250benef.mgmt.oshean.org	active



# Thats it!

---

- I know its a lot of information (these slides will all be made available)
- There is a dedicated slack channel where me (and the rest of the NAP team) are available
- You can put in a ticket and we'll get back with you
- Ultimately GNAT is fully production and the NAP team will work to resolve problems 24x7 and help you complete your maintenances