



LOGICALIS

NEP@L

Cisco NSO Integration

Edgardo Scrimaglia

NEP@L mentor

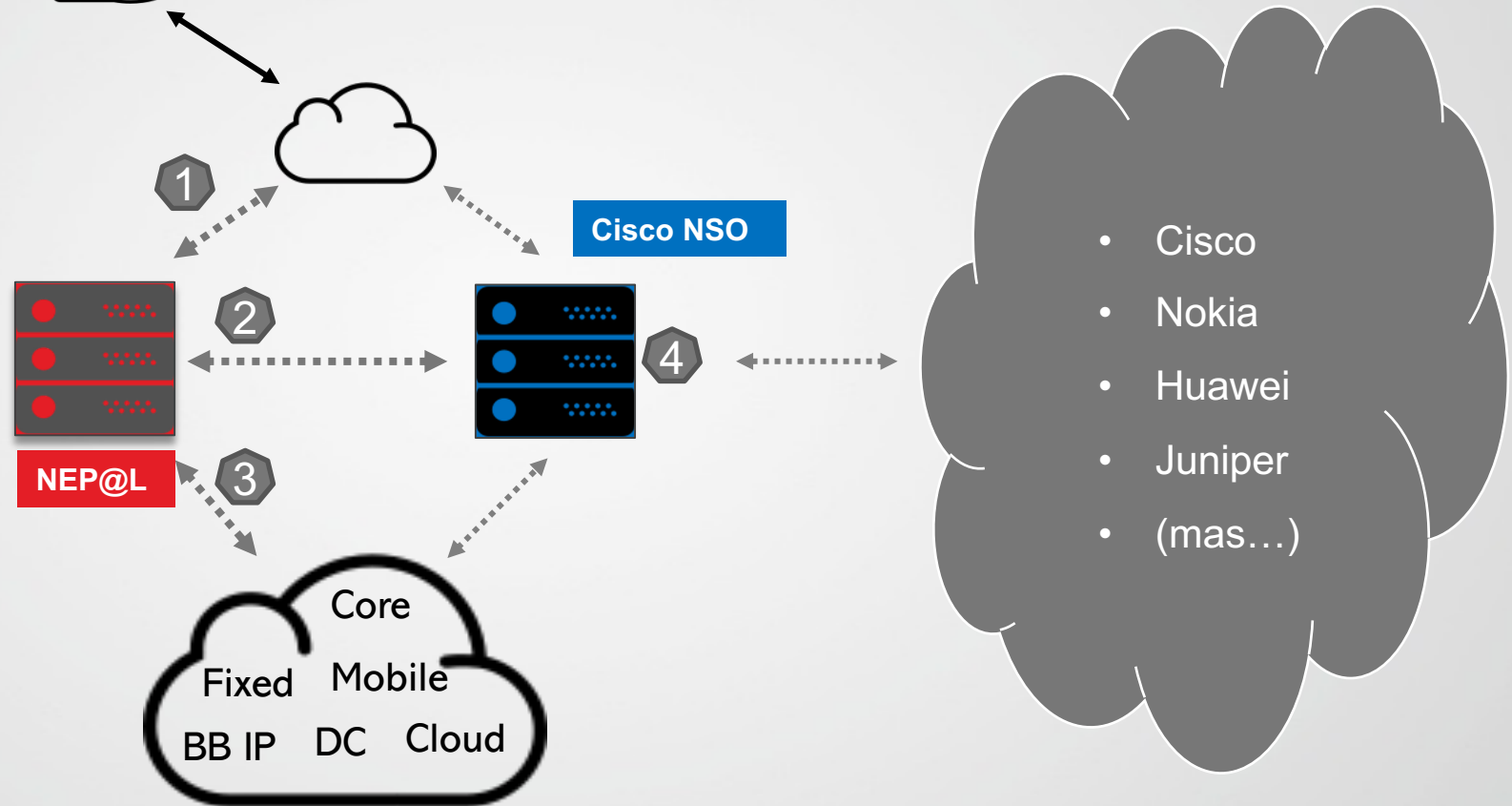
```
function ReplaceString($clientLast || $DbLast - $clientLast)
{
    $DbLast = $clientLast;
    $words = Str_Replace(' ', Str_Replace($DbLast, ' ', $clientLast));
    $explode = explode(' ', $words);
    for ($i = 0; $i < count($words); $i++) {
        if(strlen($words[$i]) > 7){
            $next = $words[$i];
        }
        else if(isset($_POST['get']))
        {
            $id = (int)$_POST['get'];
            if($id > 0)
            {
                echo json_encode($o->GetLinkData($id, false));
            }
            else if(isset($_POST['getM']))
            {
                $result = (int)$result['id'];
                echo json_encode($o->GetLinksData($_POST['getM']));
            }
            else if(isset($_POST['rating']))
            {
                function query($show, $what, $where, $parameters)
                {
                    $result = $o->RateLink($_POST['rating']);
                    if($result)
                    {
                        $show = htmlspecialchars(stripslashes(mysql_real_escape_string($show)));
                        $what = htmlspecialchars(stripslashes(mysql_real_escape_string($what)));
                        $where = "1";
                    }
                }
            }
        }
    }
}
```

NEP@L and NSO architecture

Plataformas
+
Integraciones



- 1 Integración con ecosistema de Cliente
- 2 Integración con NSO
- 3 Acceso a la Red Multi Domain -SoT
- 4 Acceso a Multi Plataformas



Modeling the integration – YANG & YAML

```
list flow-export {  
  description "Flow export service";  
  
  key device;  
  
  uses ncs:service-data;  
  ncs:servicepoint flow-export-servicepoint;  
  
  leaf device {  
    tailf:info "Target PE device";  
    tailf:cli-allow-range;  
    type leafref {  
      path "/ncs:devices/ncs:device/ncs:name";  
    }  
  }  
  
  leaf template {  
    tailf:info "Target flow-export-  
template to use";  
    type leafref {  
      path "/ncs:services/flow-export-  
template:flow-export-template/flow-export-template:name";  
    }  
  }  
}
```

NSO

From the abstraction

to an instance

```
Flow_export:  
  
  vrf:  
    - INET_MASIVO_  
    - default  
  
  devices:  
    - device: nepal-iosxr0  
      template: nepal-template  
  
  interface:  
    - id: alias-1  
      description: configuration flow-  
export template nepal-template  
      interface-iosxr:  
        GigabitEthernet: 0/0/0/2
```

NEP@L

Coding the integration – Python on NSO

```
import requests

# credentials
API_USER = 'admin'
API_PASS = 'admin'

# nso server address
API_BASE = 'http://nso:8080'

# api headers
API_HEAD = {
    'Accept': 'application/vnd.yang.data+xml'
}

api_session = requests.Session()
api_session.auth = (API_USER, API_PASS)

# create nso device from csr1.xml
api_endpoint = f'{API_BASE}/api/running/devices/device/csr1'
with open('nso_template_new_device.xml') as xml:
    api_response = api_session.put(api_endpoint, headers=API_HEAD, data=xml)
print(f' -> PUT: {api_endpoint}')
print(f' -> RESPONSE: {api_response.status_code}')
.
.
.
```

Procedural/OOP

- What to do
- How to do

Coding the integration – Ansible or Python on NEP@L

```
# Bind and Unbind Netflow SERVICE on NSO.
---
- name: "*** Bind/Unbind Template to device NSO ***"
  hosts: localhost
  connection: local
  gather_facts: no

  vars_files:
    - ../Data_Model/Service_Model.yaml

  tasks:
    - name: include var file
      include_vars:
        file: nso_vars.yaml
        name: nso

    - name: Set Facts
      set_fact:
        var_service_model: "{{Service_model}}"
        state: "{{Service_model.Parametros_Env.state}}"

    - name: Bind flow-export template to device
      nso_config:
        url: "{{nso.nso_url}}"
        username: "{{nso.nso_user}}"
        password: "{{nso.nso_pass}}"
        data:
          tailf-ncs:services/flow-export:
            - device: "{{item.device}}"
              template: "{{item.template}}"
              interface: "{{item.interface}}"
              __state: "{{state}}"
      register: salida
```

Declarative
- What to do

NEP@L Use Case definition

- New device in NSO:
 - Device Cisco IOS XR
 - Check SHH Authentication
 - Sync-from device to CDB NSO
- Create VRFs lists
- Create new templates :
 - Flow export configuration:
 - Flow receivers
 - Sampling
 - Timeouts
 - Cache entries
 - Records
- Impact de service:
 - Bind template
 - Monitoring interfaces config

NEP@L gives CI/CD to the integration

Pipeline NSO_SERVICE_NETFLOW

NEP@L CI/CD:NSO-Service-Netflow

Configuration for router flow export (Netflow, Netstream)

editar descripción

Desactivar el Proyecto

Recent Changes

Stage View

Average stage times:
(Average full run time: ~4min 20s)

#133

Sep 24 15:08

2 commits

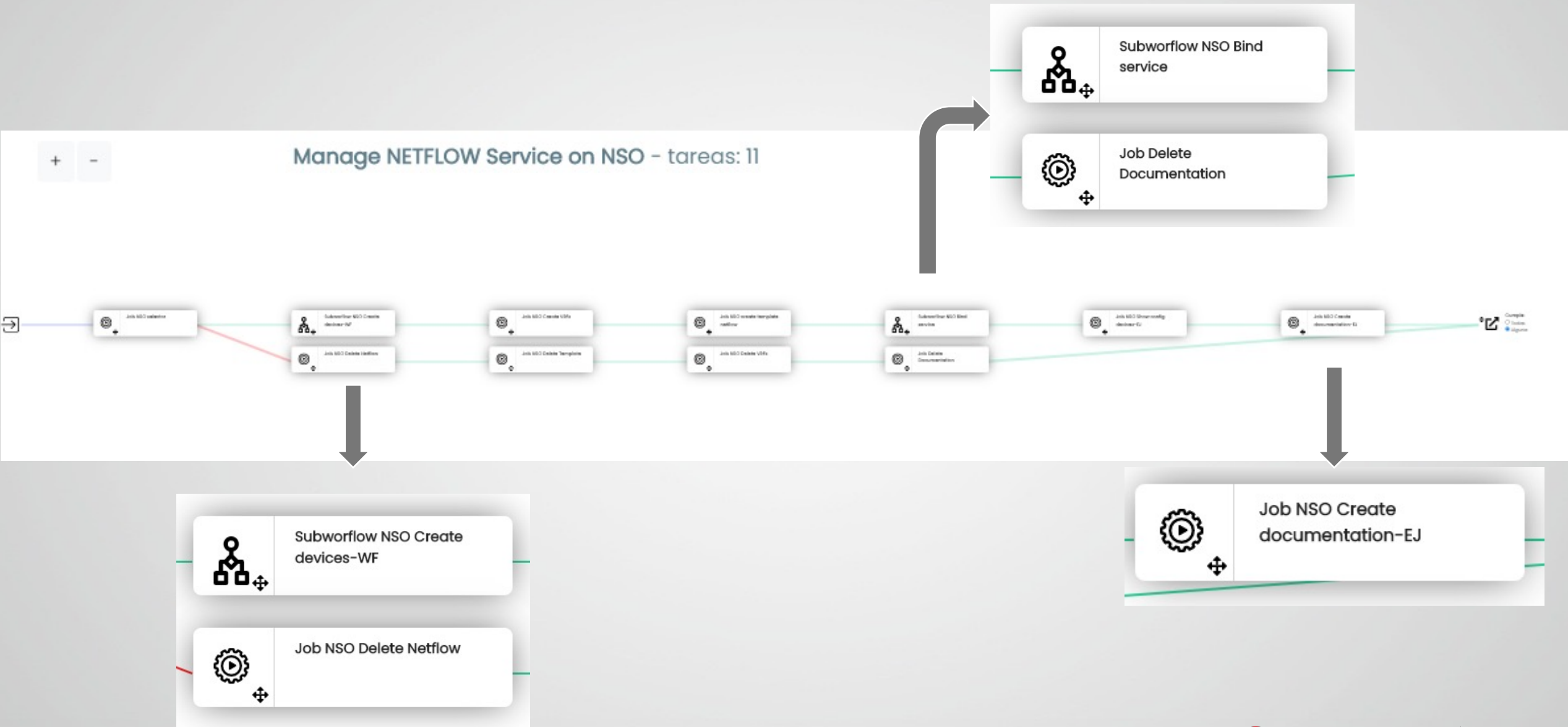
#132

Sep 24 10:28

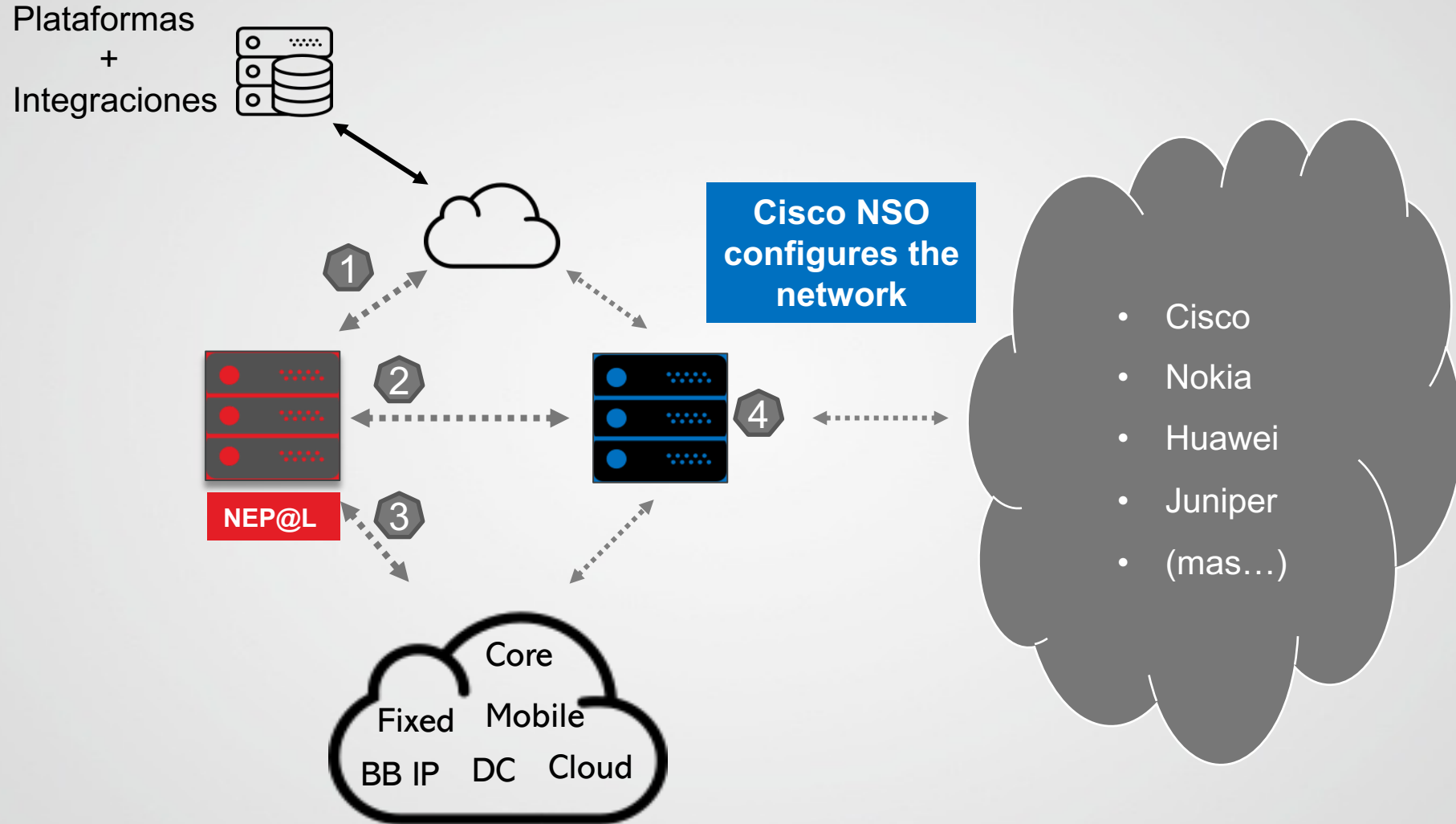
1 commit

Declarative: Checkout SCM	Code Review	Config Devices	List VRFs creation and verify	Config Netflow Template and verify	Check Devices	Show Config Devices before NSO config	NSO Deploy services	Show Actual Config Devices	Delivery	Declarative: Post Actions
3s	7s	43s	32s	27s	13s	11s	18s	10s	25s	1min 0s
4s	7s	43s	32s	27s	13s	11s	18s	10s	25s	1min 2s
3s	6s	42s	32s	27s	13s	12s	18s	10s	25s	59s

NEP@L gives a workflow manager to the integration



NSO is the Configuration Manager, NEP@L the Workflow and CI/CD





LOGICALIS

Thanks

```
function ReplaceString($clientLast || $DbLast - $clientLast)
{
    $DbLast = $clientLast;
    $words = Str_Replace(' ', Str_Replace($DbLast, ' ', $clientLast));
    $explode = explode(' ', $words);
    for ($i = 0; $i < count($words); $i++) {
        if(strlen($words[$i]) > 7){
            $next = $words[$i];
        }
        else if(isset($_POST['get']))
        {
            $id = (int)$_POST['get'];
            if($id > 0)
            {
                echo json_encode($o->GetLinkData($id, false));
            }
            else if(isset($_POST['getM']))
            {
                $result = mysql_fetch_assoc(mysql_query("SELECT MAX(id) as id FROM table"));
                echo json_encode($o->GetLinksData($_POST['getM'], $result["id"]));
            }
            else if(isset($_POST['rating']))
            {
                function query($show, $what, $where, $parameters)
                {
                    $result = $o->RateLink($_POST['linkClass'], (int)$show);
                    if($result)
                    {
                        echo "1";
                    }
                    else
                    {
                        echo "0";
                    }
                }
                $show = htmlspecialchars(stripslashes(mysql_real_escape_string($show)));
                $what = htmlspecialchars(stripslashes(mysql_real_escape_string($what)));
                $where = "1";
            }
        }
    }
}
```