

# DevOps и Автоматизация

## с Dell EMC PowerEdge

Артём Булкин  
Artem.Bulkin@dell.com



# План на сегодня

- Что такое инфраструктура как код?
- Зачем нужна инфраструктура как код?
- Как использовать IaC:
  - Встроенные средства iDRAC OME
  - Использование Python и PowerShell
  - Использование Ansible
  - Использование средств VMware

# Что такое инфраструктура как код?

# Инфраструктура как код

Infrastructure as Code (IaC) – это подход к описанию и конфигурированию инфраструктуры ЦОД с использованием конфигурационных файлов.



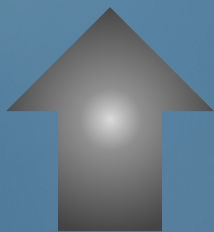
IaC позволяет:

- Повысить скорость внедрения;
- Снизить затраты на внедрение, за счёт его ускорения;
- Минимизировать риски.



# Методы IaC

Сегодня используется два метода для IaC: «Push» и «Pull».



Взаимодействие происходит с использованием REST API

## PULL

В этом варианте инфраструктура «забирает» свою конфигурацию (код) из источника



## PUSH

В этом варианте инфраструктура «получает» свою конфигурацию (код) из источника



# Почему?

# Рост требует автоматизации



Управление большими  
инфраструктурами  
(с минимизацией ошибок)



Поддержка DevOps циклов  
с **инфраструктурой как  
код**



Предоставить  
возможность  
управления  
инфраструктурой как  
в облаке



# Масштабирование ИТ-операций



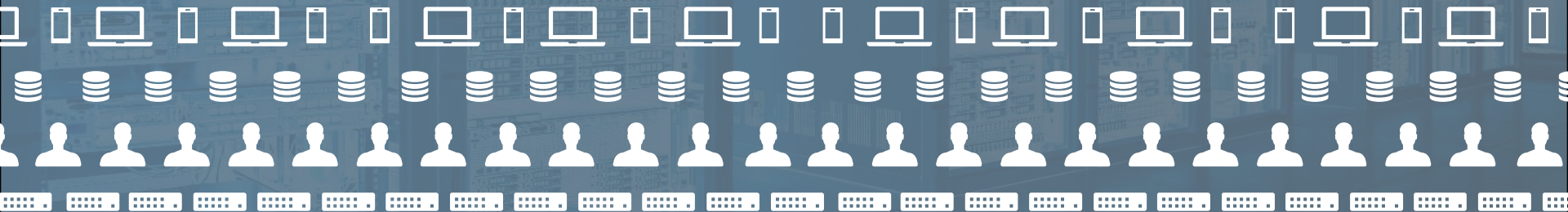
Консистентность



Безошибочность



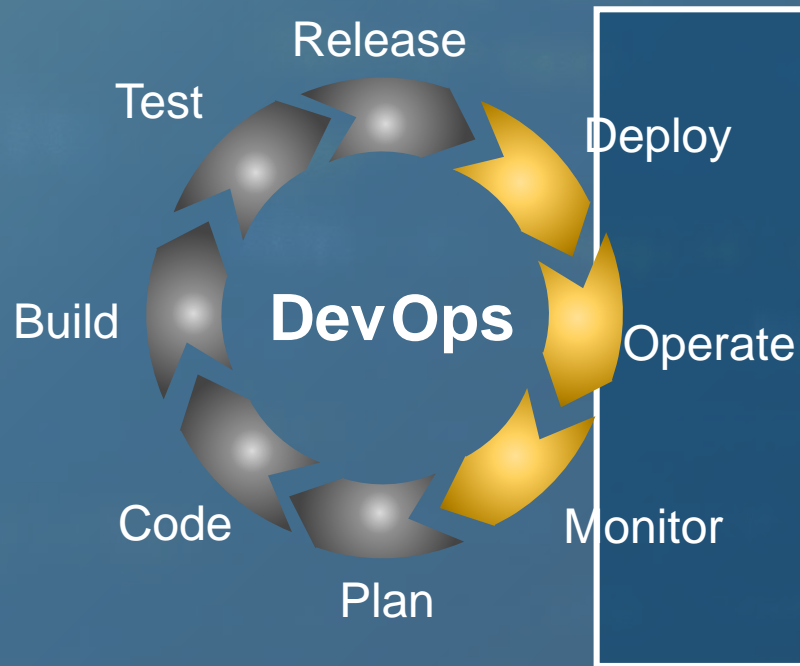
Безопасность





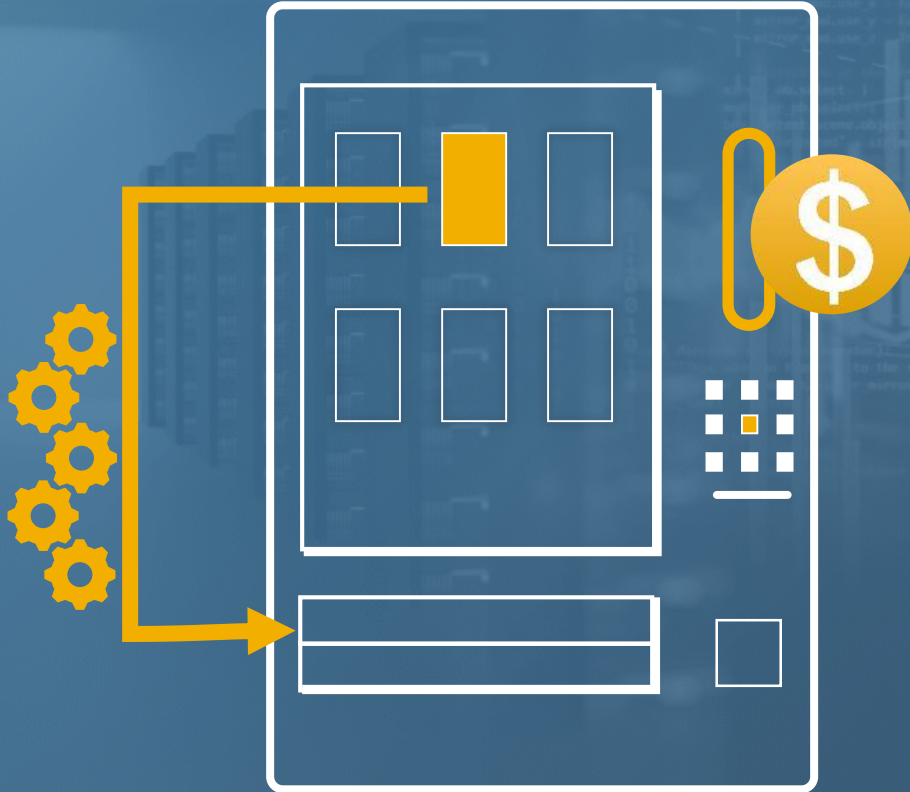
# DevOps Автоматизация

Continuous Integration | Continuous Deployment

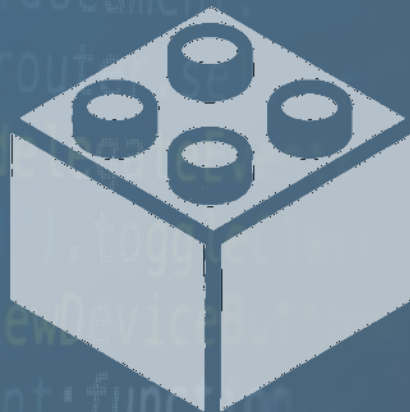


Оркестрация CI/CD  
требуется наличия  
инфраструктуры как  
кода

# Заказ ресурсов через порталы



# Автоматизация стала намного проще



## Скрипты

- Требуется навыков
- Занимает много времени
- Сложно поддерживать

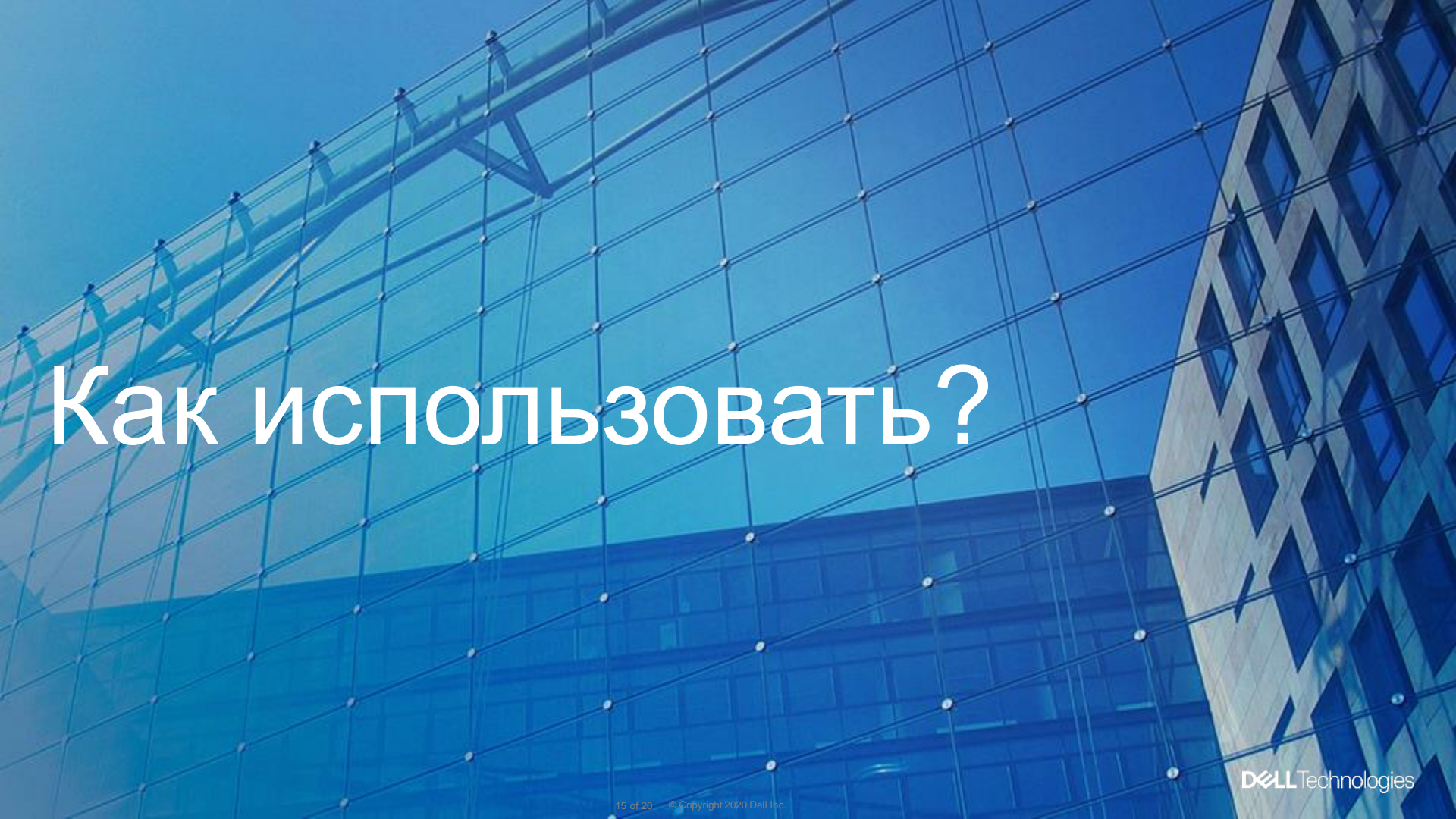
## Готовые модули для автоматизации

- Быстро внедрить
- Масштабируемые
- Легко поддерживать



# Что ещё даёт инфраструктура как код?

1. Переход к DevOps
2. Возможность использовать единые средства оркестрации в гетерогенных инфраструктурах
3. Возможность использования Agile-подхода и для управления серверной инфраструктурой:
  1. Ускорение времени принятия решений;
  2. Быстрое внесение изменений;
  3. Другие подходы к кибербезопасности;
4. Контроль версионности изменений и повышения уровня коллаборации



# Как использовать?

# Основы IaC

**Основное – это иметь конфигурационные файлы, в которых описано состояние инфраструктуры**

Server Configuration Profile (SCP) – конфигурационный файл в форматах JSON или XML, который применяется к серверу вручную (через веб-интерфейс или CLI) или с использованием средств автоматизации.

SCP можно получить:

1. Через вызовы WebGUI или CLI;
2. Вызовы Rest API;
3. OpenManage Enterprise (GUI или REST API).



## Server Configuration Profile

- BIOS
- RAID
- NIC
- Virtual I/O identities
  - Virtual MAC Address
  - WWN, WWPN
  - iSCSI Name
- iDRAC
- Network
  - VLAN
  - Type (QoS)



# Основы IaC

**Основное – это иметь конфигурационные файл, в котором описано состояние инфраструктуры**

Иметь скрипт, который последовательно меняет параметры компонент сервера.

Скрипт может использовать:

1. Внешние интерфейсы CLI:
  1. SSH
  2. RACADM;
2. Вызовы Rest API;
3. OpenManage Enterprise (REST API).



- BIOS
- RAID
- NIC
- Virtual I/O identities
  - Virtual MAC Address
  - WWN, WWPN
  - iSCSI Name
- iDRAC
- Network
  - VLAN
  - Type (QoS)

# Ресурсы

- iDRAC User Guide
  - [https://www.dell.com/support/manuals/en-us/idrac9-lifecycle-controller-v4.x-series/idrac9\\_4.00.00.00\\_ug\\_new/overview-of-idrac](https://www.dell.com/support/manuals/en-us/idrac9-lifecycle-controller-v4.x-series/idrac9_4.00.00.00_ug_new/overview-of-idrac)
- Zero Touch Bare Metal Deployment WhitePaper
  - [https://downloads.dell.com/manuals/all-products/esuprt\\_software/esuprt\\_it\\_ops\\_datcentr\\_mgmt/dell-management-solution-resources\\_white-papers9\\_en-us.pdf](https://downloads.dell.com/manuals/all-products/esuprt_software/esuprt_it_ops_datcentr_mgmt/dell-management-solution-resources_white-papers9_en-us.pdf)
- iDRAC CLI Reference Guide
  - [https://dl.dell.com/topicspdf/idrac9-lifecycle-controller-v4x-series\\_cli-guide\\_en-us.pdf](https://dl.dell.com/topicspdf/idrac9-lifecycle-controller-v4x-series_cli-guide_en-us.pdf)
- iDRAC 9 Redfish API Guide
  - [https://dl.dell.com/topicspdf/idrac9-lifecycle-controller-v4x-series\\_api-guide\\_en-us.pdf](https://dl.dell.com/topicspdf/idrac9-lifecycle-controller-v4x-series_api-guide_en-us.pdf)

# Использование OpenManage Enterprise

- Внедрение с использованием шаблонов, создаваемых на основе серверов
- Шаблоны можно править в любой момент времени
- Шаблоны применяются к выбранным серверам или группам серверов, применяя к ним указанные параметры физических и виртуальных устройств

Клонировать или импортировать профиль с существующего сервера



Deployment is done by associating a profile with a slot or physical server

## Server Configuration Profile

- BIOS
- RAID
- NIC
- Virtual I/O identities
  - Virtual MAC Address
  - WWN, WWPN
  - iSCSI Name
- iDRAC
- Network
  - VLAN
  - Type (QoS)

Применяется к серверу или слоту



1:1



1:many



Device Group(s)



Remote sites



# Интеграция со сторонними средствами

## Просто

- Расширение Redfish
- Модель данных JSON

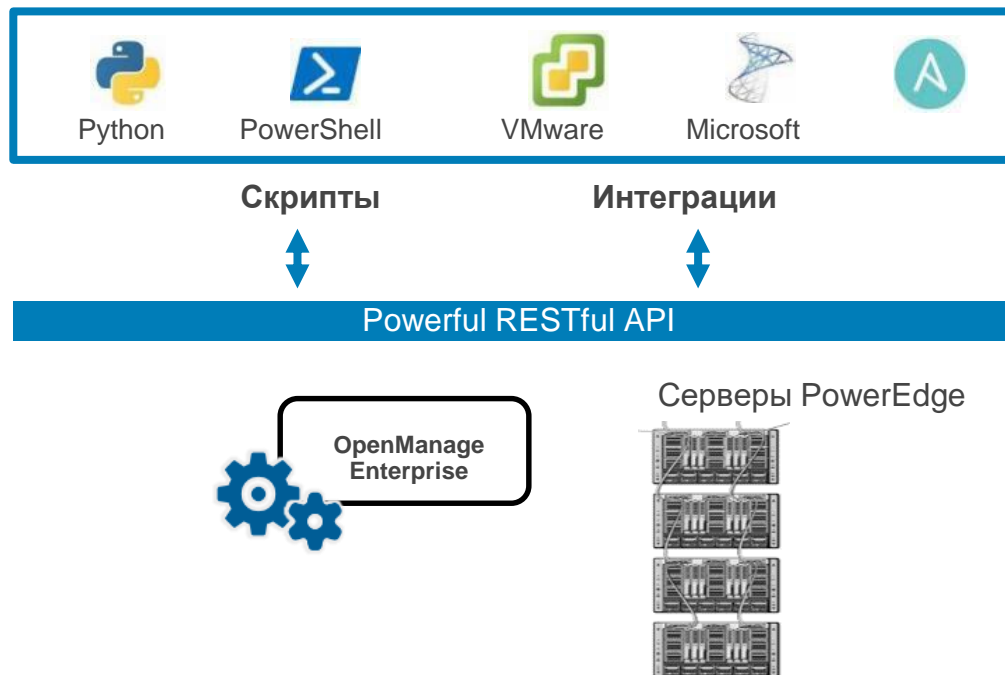
## Безопасно

- Использует SSL-шифрование

## Мощно

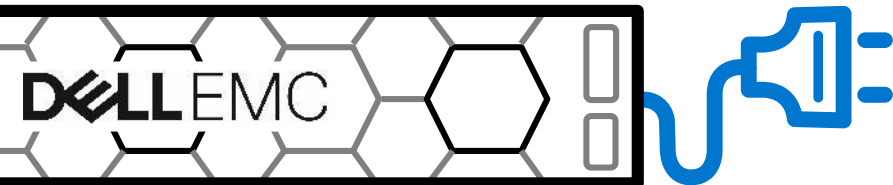
- Использует 100% функционала
- Поддержка разных сценариев внедрения

Comprehensive REST API to integrate with other tools



Click [here](#) for API resource

# Сторонние инструменты



## Типы нагрузок



VMs



Containers  
(Kubernetes)



Enterprise Apps

## Автоматизация



vmware  
vRO



Ansible



Python



PowerShell

## Платформы IT-Ops



OpenStack



vmware  
vRA

servicenow

DELL Technologies



</Python>



Поддерживается  
сообществом

## Workload Support



VMs



Containers  
(Kubernetes)



Enterprise Apps

## Automation



vmware  
vRO



Ansible



Python



PowerShell

## IT-Ops Platforms



OpenStack

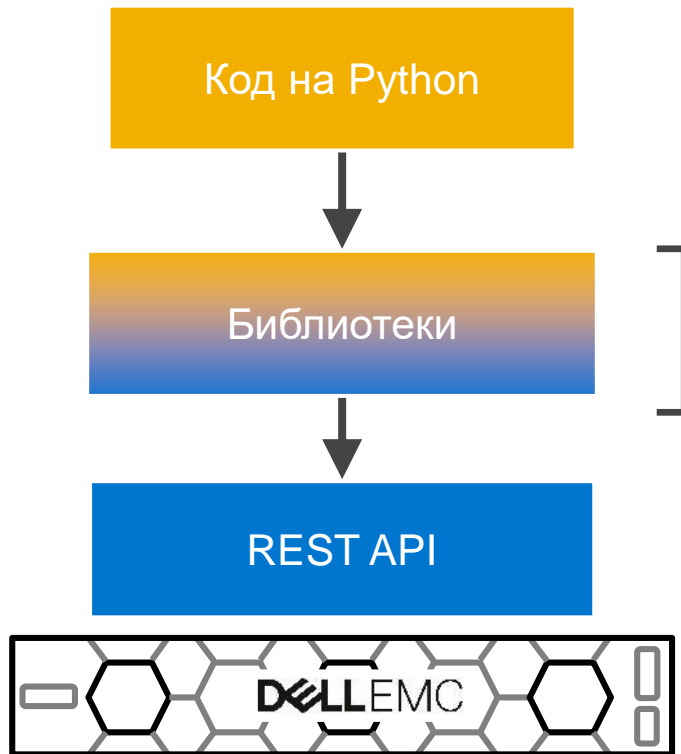


vmware  
vRA

servicenow

DELL Technologies

# Интеграция с Python



## Python Libraries

Пишутся самостоятельно, или можно использовать готовые наборы





</PowerShell>



Поддерживается  
сообществом

## Workload Support



VMs



Containers  
(Kubernetes)



Enterprise Apps

## Automation



vmware  
vRO



Ansible



Python



PowerShell

## IT-Ops Platforms



OpenStack



vmware  
vRA

servicenow

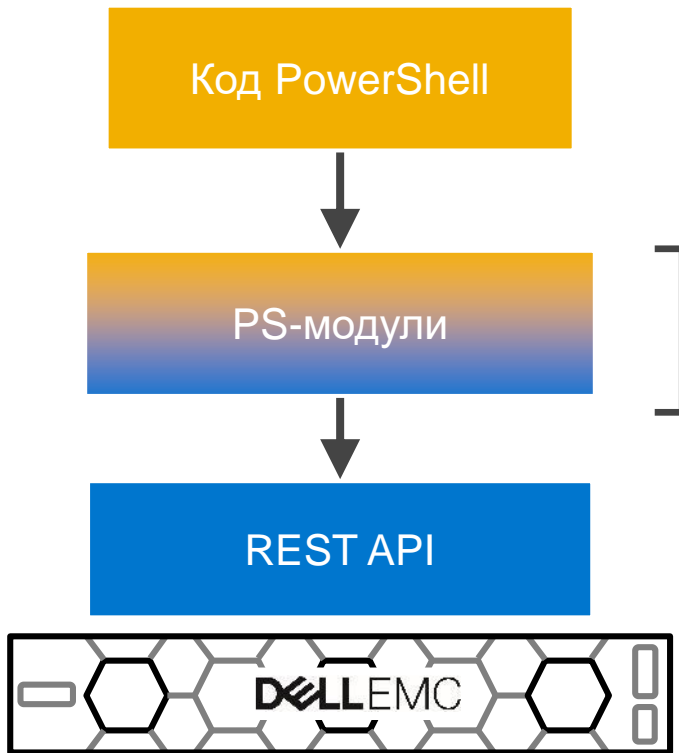
DELL Technologies

# PowerShell



- “CLI” на стероидах для Windows
- Простой и мощный инструмент
  - Структурируемый и читаемый
  - Автозаполнение
  - PS ISE – полноценный скриптовый язык
- **PowerShell Core** работает на Linux и Mac!

# Интеграция с PowerShell



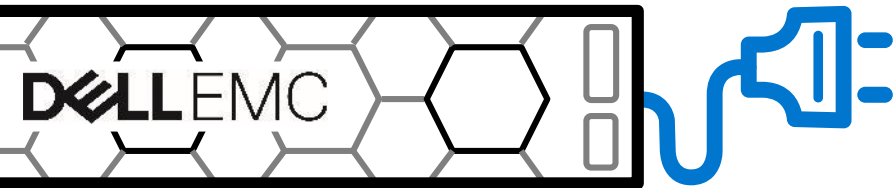
## PowerShell Packages

Пишется самостоятельно или  
используются готовые модули

# Ссылки на ресурсы

- PowerShell Tools for Dell EMC PowerEdge:
  - <https://github.com/dell/DellPEWSMANTools>
- Dell EMC OpenManage Python SDK:
  - <https://github.com/dell/omsdk>
- iDRAC Redfish Scripting
  - <https://github.com/dell/iDRAC-Redfish-Scripting>
- Dell на Github
  - <https://github.com/dell>
- Knowledge Base – Redfish API
  - <https://www.dell.com/support/article/en-us/sln310624/redfish-api-with-dell-integrated-remote-access-controller?lang=en>
- OpenManage Enterprise Python and Powershell scripts
  - <https://github.com/dell/OpenManage-Enterprise>





</Ansible Modules>

## Workload Support



VMs



Containers  
(Kubernetes)



Enterprise Apps

## Automation



vmware  
vRO



Ansible



Python



PowerShell

## IT-Ops Platforms



OpenStack



vmware  
vRA

servicenow

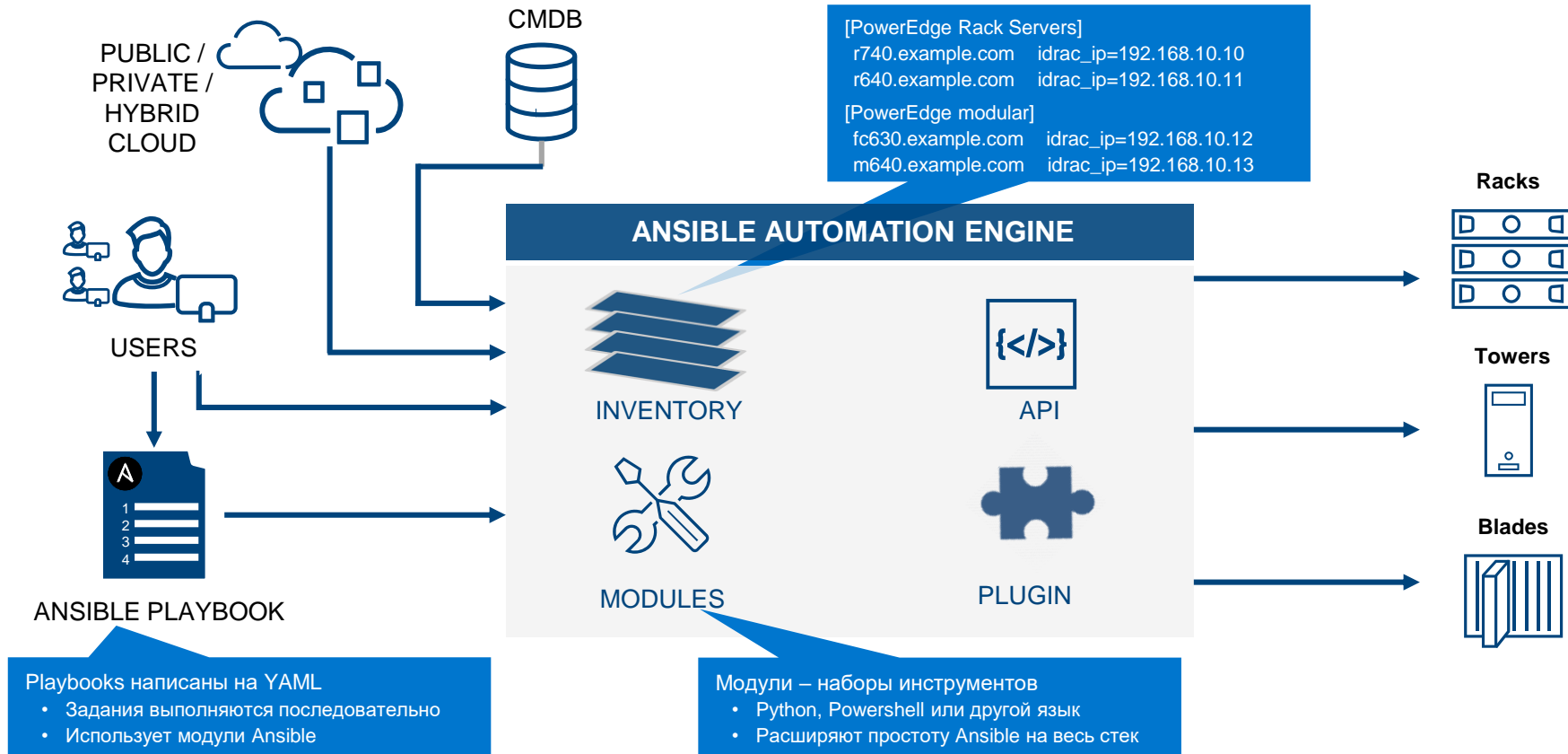
DELL Technologies

# Ansible – автоматизации инфраструктуры и нагрузок



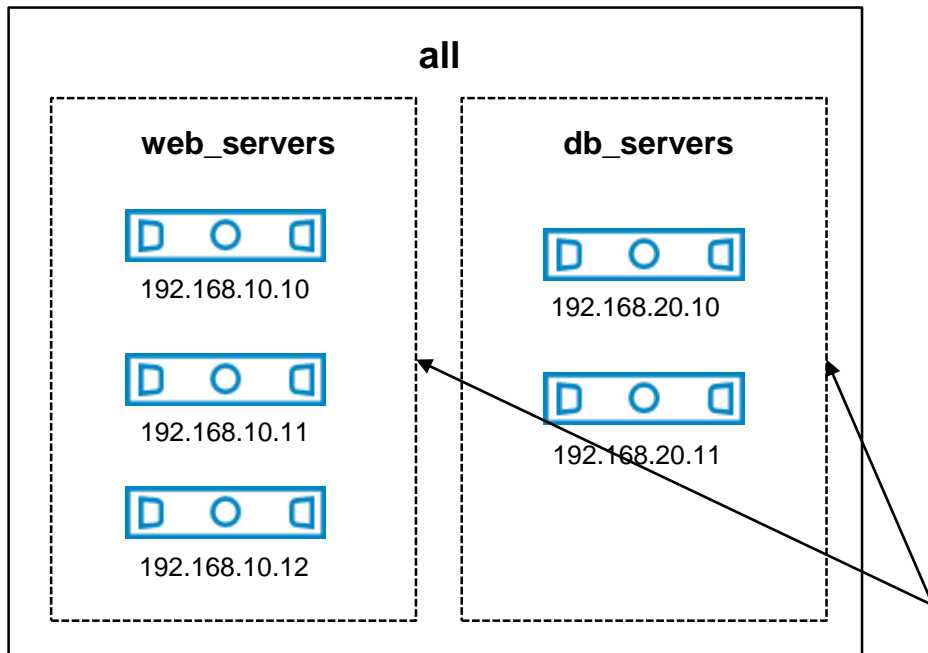
- **Мощный движок автоматизации**
  - Выделение и управление
  - Развёртывание ОС и приложений
  - Управление серверами, СХД и коммутаторами
  - Управление серверами, ВМ и приложениями
  - Применение патчей и обновлений
- **Простая настройка**
  - Использование YAML
  - Выполнение без агентов
- **OpenSource, поддерживается RedHat**
  - Ansible Tower – полностью поддерживаемый коммерческий вариант
  - AWX – бесплатная версия Tower

# Как работает Ansible?



# Inventory

Inventory – это файл, содержащий информацию о целевых устройствах, и может быть задан в любом формате (ini, YAML и т.д.)



## inventory.ini

---

### [web\_servers]

```
web_server_01 ansible_host=192.168.10.10
web_server_02 ansible_host=192.168.10.11
web_server_03 ansible_host=192.168.10.12
```

### [db\_servers]

```
db_server_01 ansible_host=192.168.20.10
db_server_02 ansible_host=192.168.20.11
```

Целевые узлы могут быть сгруппированы по назначению или ЦОД

# Автоматизация с Ansible Playbooks

Playbook написаны на YAML и используются для конфигурирования или развёртывания

## • Target

- Список групп или хостов
- Целевые платформы

## • Tasks

- «Минимальное» задание
- Исполняются по порядку, одну за раз

## • Plays

- Каждый “play” содержит набор заданий
- Все хосты получают одни и те же задания

## • Playbook

- Содержит несколько «play»

```
- hosts: poweredge_rack_servers
  tasks:
```

Target

```
- name: GET SYSTEM INVENTORY
```

Task Name

```
  dellemc_get_system_inventory:
```

```
    idrac_ip: "{{ inventory_hostname }}"
```

```
    idrac_user: "{{ username }}"
```

```
    idrac_password: "{{ password }}"
```

Module

```
- name: GET FIRMWARE INVENTORY
```

```
  dellemc_get_firmware_inventory:
```

```
    idrac_ip: "{{ inventory_hostname }}"
```

```
    idrac_user: "{{ username }}"
```

```
    idrac_password: "{{ password }}"
```

Переменные

```
- hosts: poweredge_modular_servers
  tasks:
```

```
- name: CONFIGURE BOOT MODE to UEFI
```

```
  dellemc_configure_bios:
```

```
    idrac_ip: "{{ inventory_hostname }}"
```

```
    idrac_user: "{{ username }}"
```

```
    idrac_password: "{{ password }}"
```

```
    boot_mode: "Uefi"
```

Module

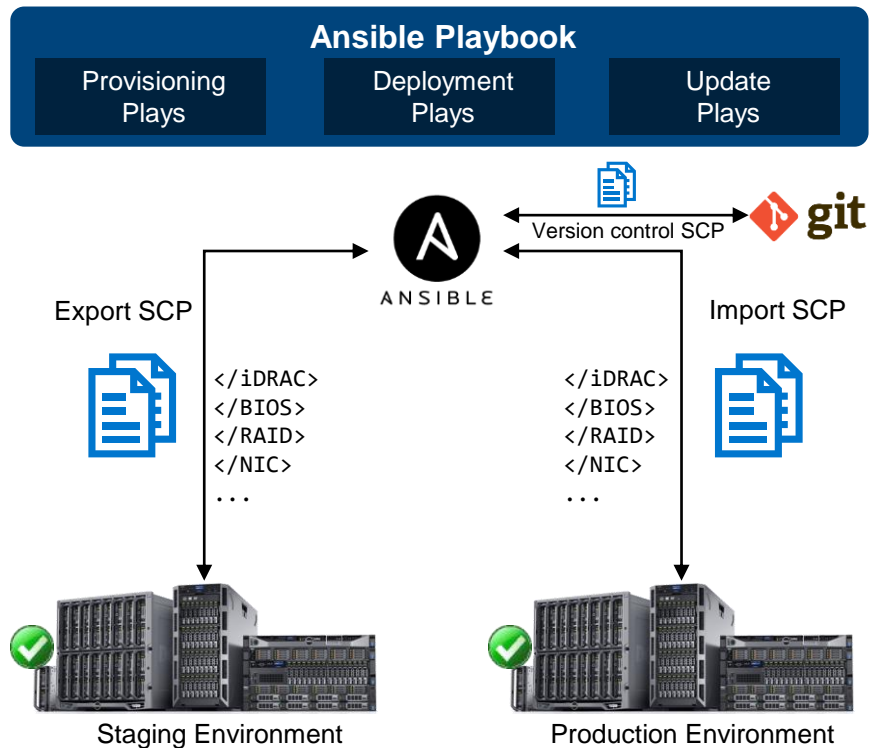
Module Arguments



# Автоматизация с iDRAC

```
- hosts: staging
  tasks:
    - name: Export SCP from a reference server
      idrac_server_config_profile:
        idrac_ip: "{{ inventory_hostname }}"
        idrac_user: "{{ username }}"
        idrac_password: "{{ password }}"
        share_name: "{{ playbook_dir }}"
        export_type: "clone"
      register: result

- hosts: production
  tasks:
    - name: Import SCP to target servers
      idrac_server_config_profile:
        idrac_ip: "{{ inventory_hostname }}"
        idrac_user: "{{ username }}"
        idrac_password: "{{ password }}"
        share_name: "{{ playbook_dir }}"
        scp_file: "{{ result.scp_status.file | basename }}"
        command: "import"
```



# Развёртывания шаблона

- hosts: localhost
- tasks:
  - name: Create a deployment template
  - ome\_template:
    - hostname: "{{ ome\_hostname }}"
    - username: "{{ ome\_username }}"
    - password: "{{ ome\_password }}"
    - command: "create"
    - device\_service\_tag: "{{ reference\_service\_tag }}"
    - attributes:
      - Name: "{{ reference\_template\_name }}"
      - Description: "Deployment template"
      - TypeId: 2
      - ViewTypeId: 2
      - Fqdds: "all"
- name: Deploy template
- ome\_template:
  - . . .
  - command: "deploy"
  - template\_name: "{{ reference\_template\_name }}"
  - device\_service\_tag: "{{ target\_service\_tags }}"

The image displays two screenshots of the 'Create Template' wizard interface, likely from a Dell iDRAC or similar management tool.

**Top Screenshot: General Tab**

- Create Template** (Title)
- Template Information** (Section Header)
- General** (Tab)
- Reference Device** (Field)
- Create Template** (Button)
- Device Selection** (Section Header)
- Only one device can be selected as a reference device** (Note)
- 1 Device Selected** (Text)
- Selected Device:** WIN-2G7L200Q2F6
- Configuration Elements** (Section Header)
- Elements to Clone** (Text)
- ☒ iDRAC
- ☒ BIOS
- Deploy Template: Deploy MX740c from Gold Master** (Text)

**Bottom Screenshot: Select Devices Tab**

- Create Template** (Title)
- Target** (Section Header)
- Boot to Network ISO** (Text)
- Target Attributes** (Text)
- Schedule** (Text)
- Select Devices** (Section Header)
- Warning! This operation is potentially destructive and is recommended for bare metal devices only.** For production devices, use the template compliance remediation workflow.
- 1 Device(s) Selected** (Text)
- ☒ Use strict checking  
Strict checking will attempt to match VLANs with like characteristics. If disabled, only VLAN name and QoS are used for matching. This option applies to MX sleds only.
- Host OS Reboot Options (if reboot is required)**
  - ☐ Do not forcefully reboot the host OS if the graceful reboot fails
- Step 1 of 4** (Text)
- Next** (Button)
- Finish** (Button)
- Cancel** (Button)

# Ресурсы

- **GitHub repository**
  - <https://github.com/dell/dellemc-openmanage-ansible-modules>
- **Documentation**
  - <https://github.com/dell/dellemc-openmanage-ansible-modules/tree/devel/guides>
- **Dell Knowledge Base**
  - <https://www.dell.com/support/article/sln310720/dell-emc-openmanage-ansible-modules>
- **Demo Center Interactive Lab:**
  - <https://interactivedemos.democenter.dell.com/v2/OMAM/index.html>
- **Dell EMC Open Source Redfish PowerShell and Python Scripting**
  - <http://github.com/dell/iDRAC-Redfish-Scripting>
- **OpenManage Enterprise Python and Powershell scripts**
  - <https://github.com/dell/OpenManage-Enterprise>



</vRO>

</vRA>

</ServiceNow>

## Типы нагрузок



VMs



Containers  
(Kubernetes)



Enterprise Apps

## Автоматизация



Ansible



Python



PowerShell

## Платформы IT-Ops



OpenStack



Dell Technologies

# vRealize Orchestrator: Drag and Drop

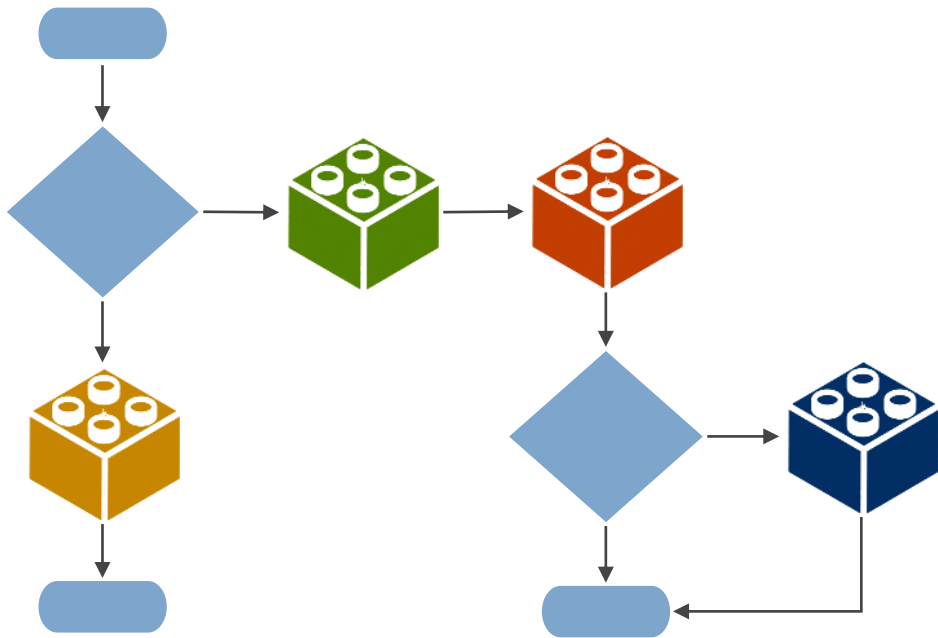
PowerMax provisioning ☆ RUN DEBUG VALIDATE ALL RUNS DELETE

Summary Variables Inputs/Outputs Schema Input Form Version History Audit

Generic

- Scriptable task
- Decision
- Decision activity
- User interaction
- Waiting timer
- Waiting event
- End workflow
- Throw exception
- Workflow note
- Action element
- Workflow element
- Foreach element

# vRO для автоматизации всего

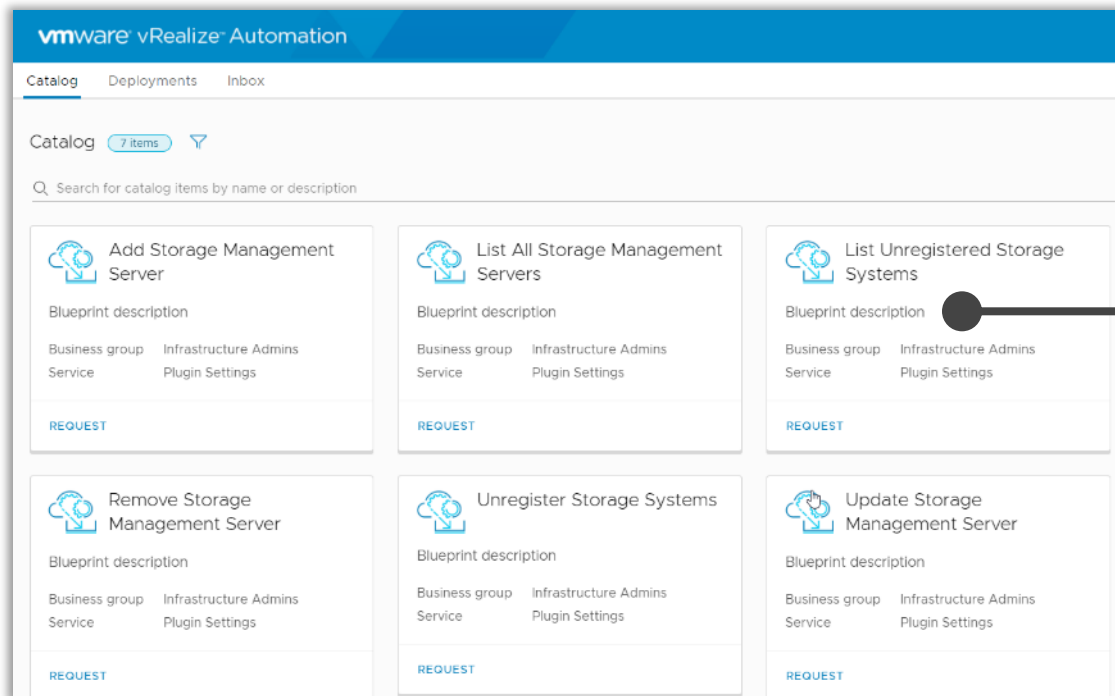


## Блочные диаграммы

Использование логических  
блоков для описания  
процессов развёртывания  
и автоматизации



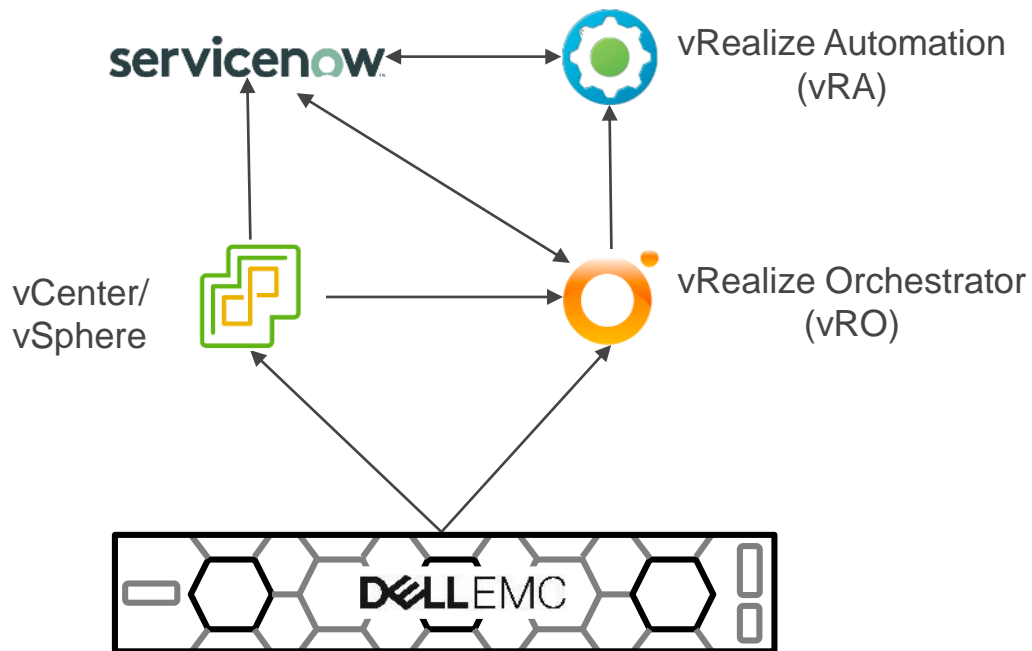
# vRealize Automation



## Порталы самообслуживания

Готовые шаблоны vRO  
можно использовать как  
элементы портала  
самообслуживания

# Различные варианты построения автоматизации



DELLEMC