

# Автоматизация процессов

с помощью решений VMware

Сергей Калугин

Cloud Solution Engineer,  
VMware



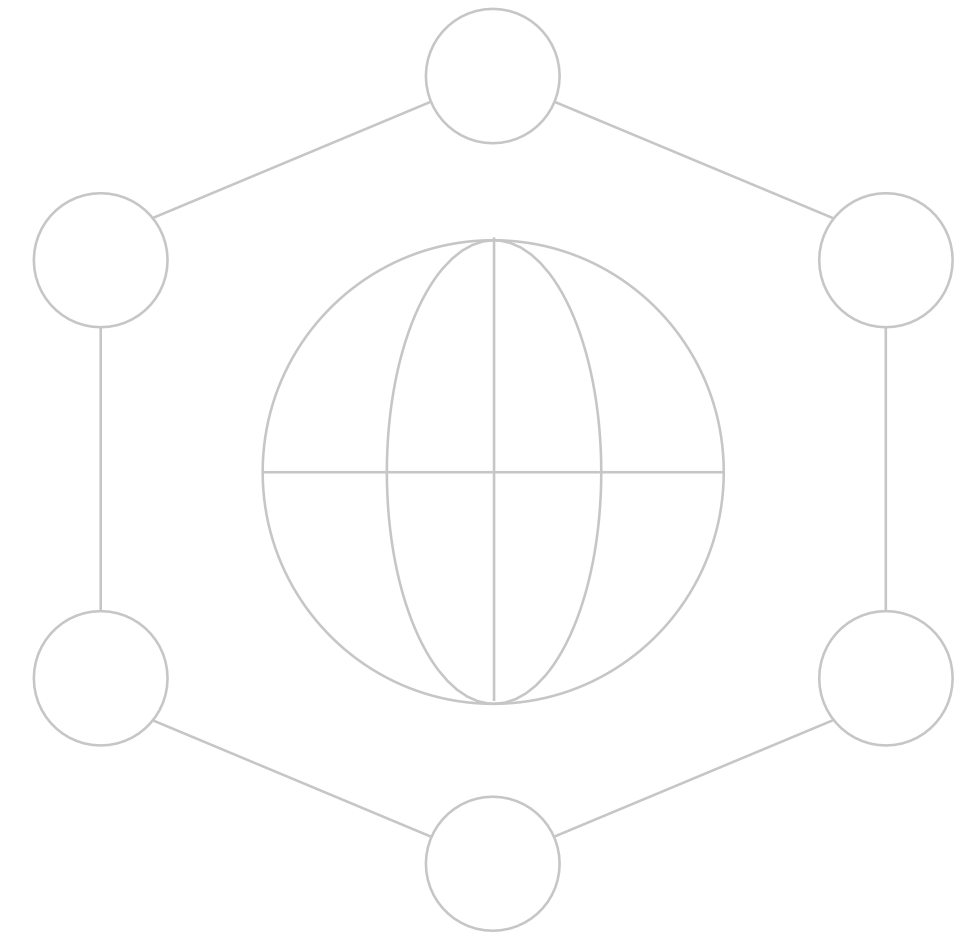
**Cloud** Thinking



# Облачные свойства

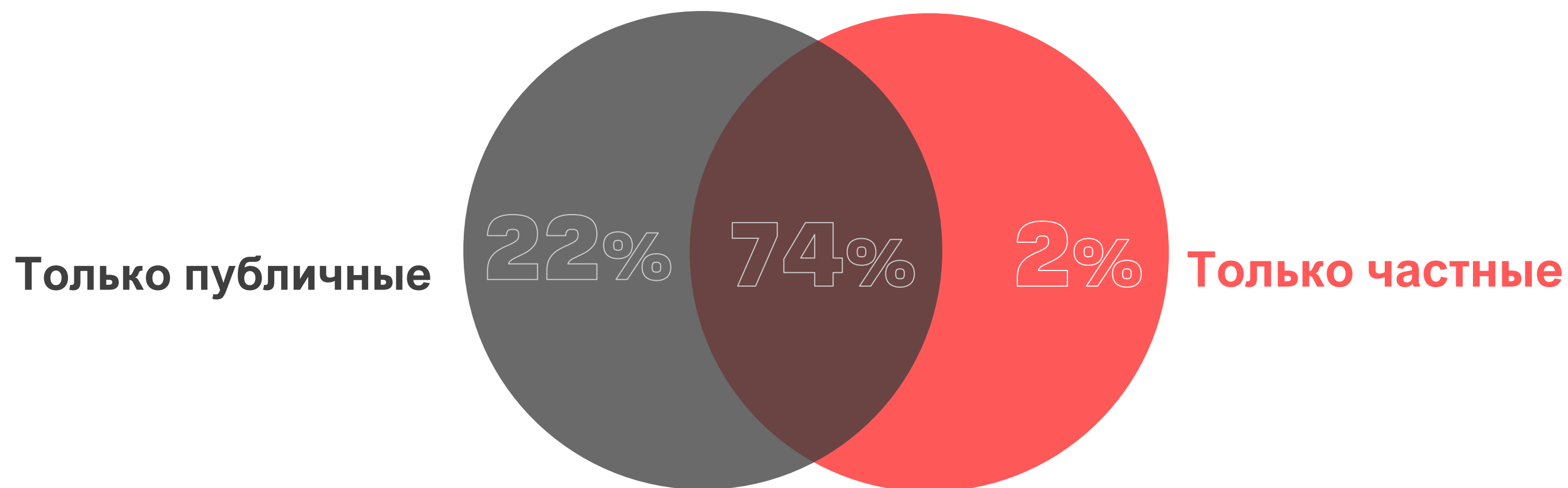
NIST, сентябрь 2011

- 01 Самообслуживание по требованию (On-demand self-service)
- 02 Широкий сетевой доступ (Broad network access)
- 03 Объединение ресурсов в пулы (Resource pooling)
- 04 Мгновенная эластичность (Rapid elasticity)
- 05 Измеряемость (Measured service)



# Как сейчас обстоят дела с облаками?

Гибридные облака «побеждают»



Публичные = 96%

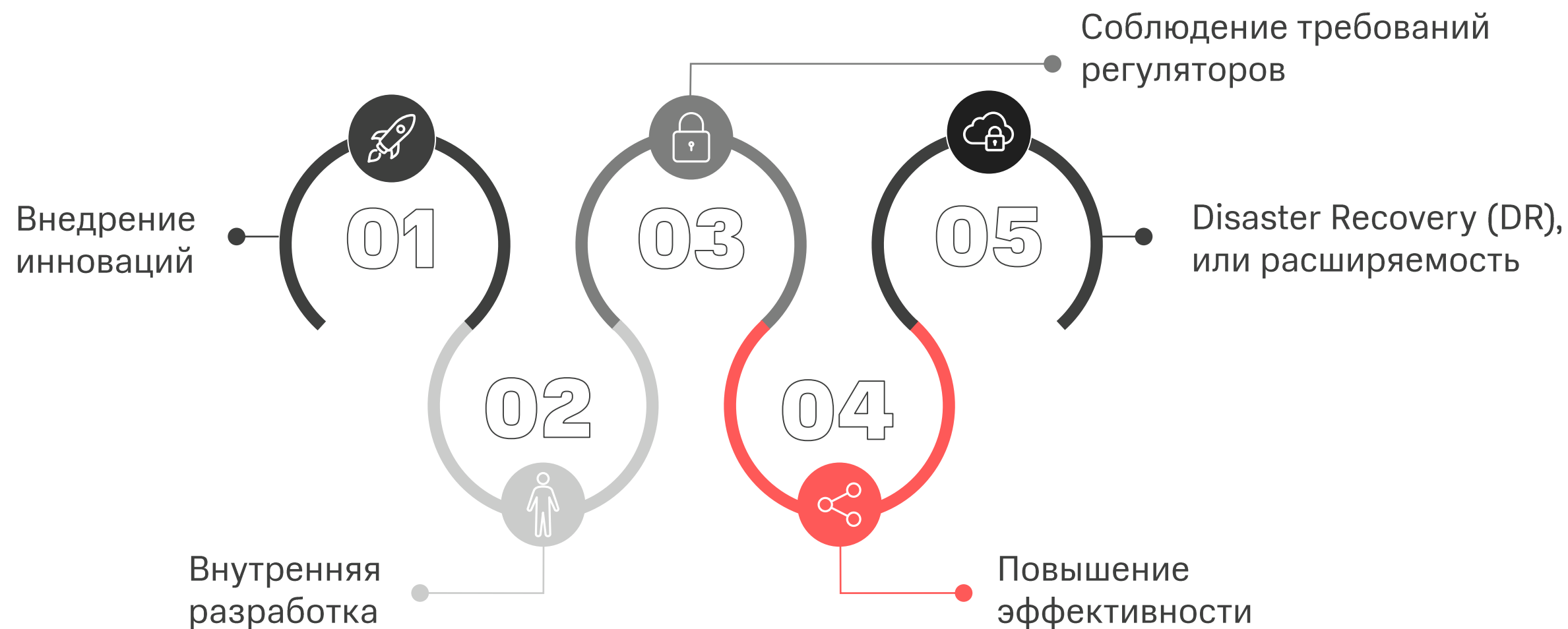
**Гибридные**

Частные = 76%

Источник: Flexera 2020 State of Cloud Report

# Почему все хотят в облака?

## 5 причин выбора мультиоблачной стратегии



# Бизнесу необходимо постоянное увеличение скорости

## 5 причин выбора мультиоблачной стратегии

### Владельцы бизнеса

Как можно быстрее предоставить новые возможности безопасным и управляемым способом



### Разработчики

Фичи = хорошо  
Контролируемо = отлично  
Вывод в прод = максимально быстро  
Fail faster, Fail cheaply



Фичи = хорошо  
Контролируемо = отлично  
Вывод в прод = максимально быстро  
Fail faster, Fail cheaply

# Кто такие разработчики?



Реализуют  
бизнес-требования



Измеряются  
реализуемыми фичами



Ничего не хотят знать  
об инфраструктуре

# Чего же они хотят?



Мгновенной  
удовлетворенности



Повторяемых процессов



Гибкости в определении,  
что им нужно

# О чем они не хотят думать?



О соответствии  
корпоративным стандартам



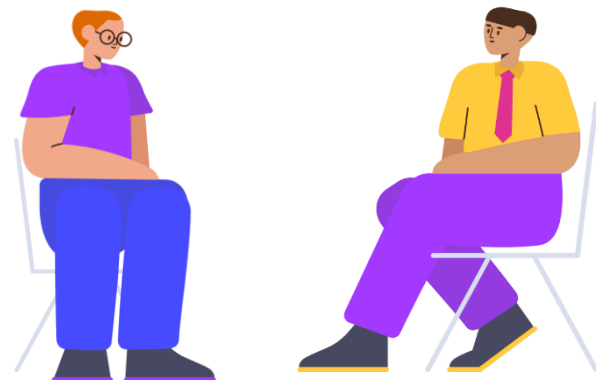
О бюджете (пока он есть)



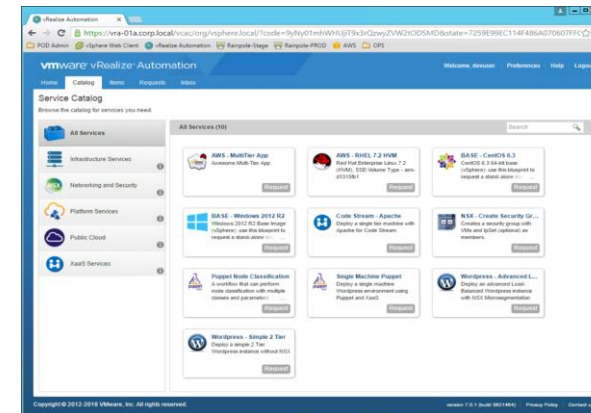
О «вечной»  
инфраструктуре



# Мы все прошли через это



«Я слышал, ты можешь помочь развернуть новое окружение для тестирования моего нового сервиса»

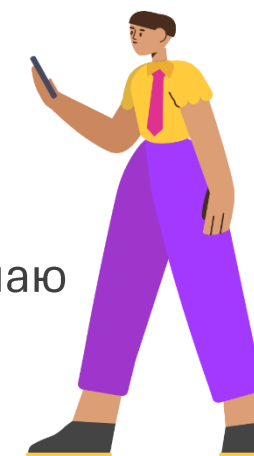


«Конечно, приятель! У нас есть классный портал самообслуживания для этих целей!»



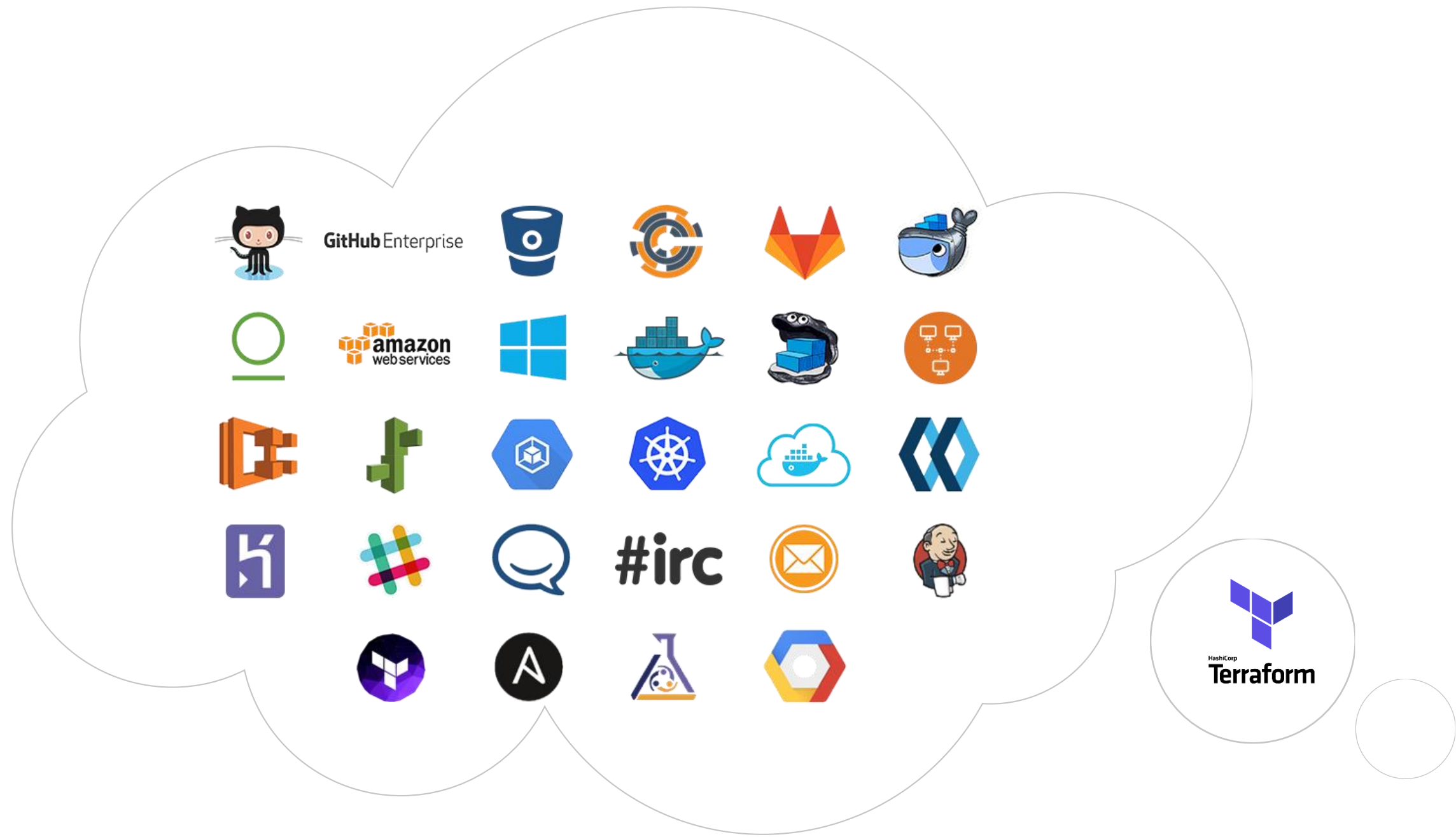
«Хм, но ничего из этого не подходит для моих задач...»

«Не дрейфь! Поставь заявку в ServiceDesk — и я все сделаю в течение 4–6 недель!»



«Да ну, забудь...»





# Terraform и Ansible!

Популярные инструменты автоматизации инфраструктуры



Для деплоя  
инфраструктуры



Для автоматизации  
конфигурации приложений

# Вкратце: Terraform и Ansible

Оба инструмента обладают следующими характеристиками:

**01**

Мультиоблачные  
возможности

**02**

Простая, легковесная  
архитектура

**03**

Просто писать шаблоны  
на основе YAML

**04**

Могут выполнять команды  
на удаленных VM

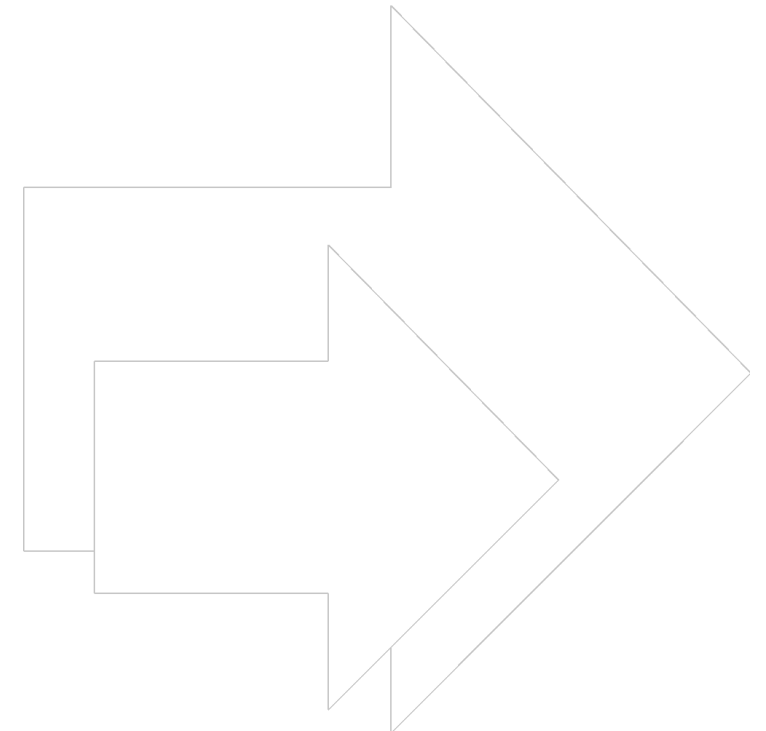
**05**

Можно использовать  
в CI-/CD-конвейерах

**Зачем искать  
что-то еще?**

# Чего не хватает для полноценного облака?

1. Настоящей «кросс-облачности».
2. Возможности делегирования задач пользователю: скрипты запускают теперь уже не «админы виртуализации», но DevOps-команда.
3. Контроля ресурсов: пользователь/скрипт может завалить запросами все имеющиеся ресурсы.
4. Отчетности: кто, когда и что разворачивал/изменял.
5. Гранулярности: все должны быть «админами».
6. Управления (Governance): согласования, оповещения.
7. Выделения ресурсов на время (аренда).
8. Планирования потребления.
9. Инструмента для «обычного пользователя» – портала самообслуживания.
10. Интеграции с корпоративными системами.
11. Биллинга.
12. И многого другого...



# Облако 2.0

- Интегрированное программное и аппаратное решение на основе гиперконвергентной архитектуры.
- Ориентировано на приложения, а не на ИТ.
- Проще разворачивать.
- Может конкурировать по стоимости с публичными облаками.
- Упрощена процедура перехода на новые версии.
- Предоставляют не только услуги IaaS, но и PaaS.
- Современные частные облака — это платформы с API.

Инфраструктура

**Сервисы**



**А если я скажу  
тебе, что...**

**ЭТО СУЩЕСТВУЕТ!**

# Платформа управления гибридным облаком

Единые операции

## Автономное управление

Оптимизация  
производительности

Управление емкостью  
и стоимостью

Мониторинг  
и устранение проблем

Соответствие  
регулятору

Управление  
журналами

Безопасность  
и прозрачность сети

## Программное развертывание

Инфраструктура  
как код

Программные конвейеры

Управление k8s

Каталог  
сервисов

Сеть  
и безопасность

Маркетплейс

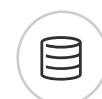
## Общая платформа

## Единая инфраструктура

### Программный ЦОД



Вычисление



Хранение



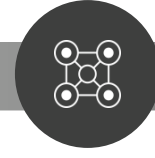
Сеть



Периферия



ЦОД



Облака локальных провайдеров

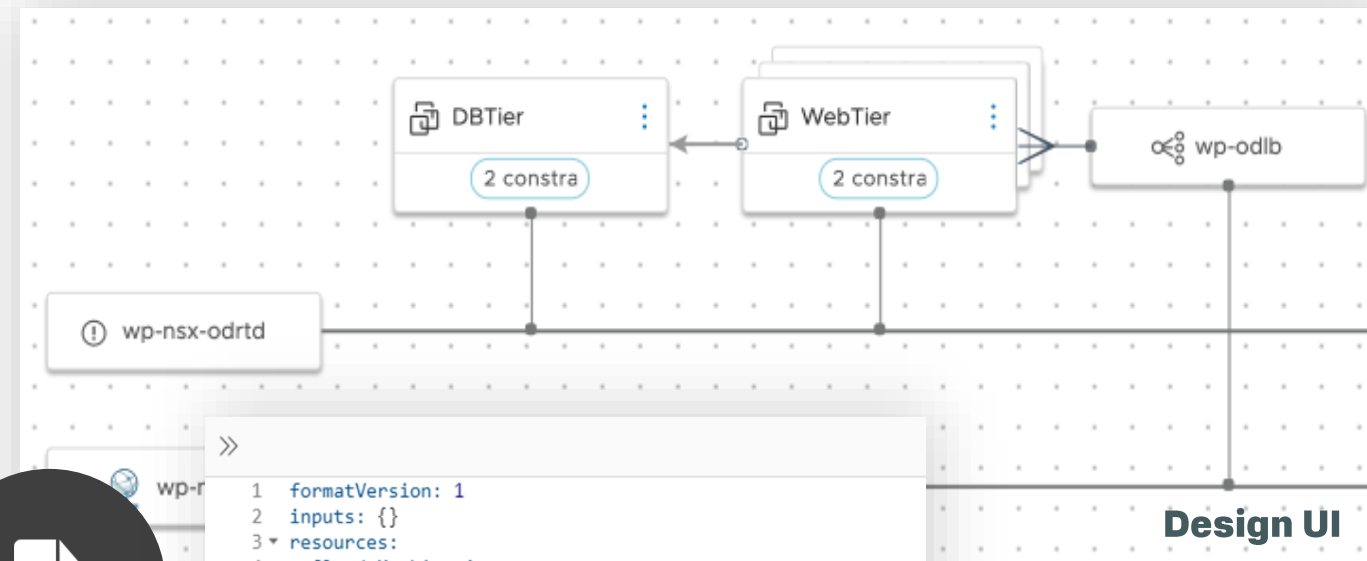


Гиперскейлеры



# Инфраструктура как код

```
1 name: wp-virtualjad-dev
2 version: 1.1
3 description: WP (latest) for virtualjad.com blog
4 inputs: ...
34 resources:
35 DBTier:
36   type: Cloud.Machine
37   properties:
38     storage:
39     constraints:
40     - tag: 'storage:${input.storage}'
41     name: wp-db
42     image: '${input.image}'
43     flavor: '${input.size}'
44     constraints:
45     - tag: 'env:ezlab'
46     - tag: 'stage:dev'
47     networks:
48     - name: '${resource["wp-nsx-odrtd"].name}'
49     cloudConfig: | ...
63 WebTier:
64   type: Cloud.Machine
65   dependsOn:
66   - DBTier
67   properties:
68     storage:
69     constraints:
70     - tag: 'storage:${input.storage}'
71     name: wp-web
72     image: '${input.image}'
73     count: 2
74     constraints:
75     - tag: 'env:ezlab'
76     - tag: 'stage:dev'
77     flavor: '${input.size}'
78     networks:
79     - name: '${resource["wp-nsx-odrtd"].name}'
80     cloudConfig: | ...
102 wp-odlb:
```



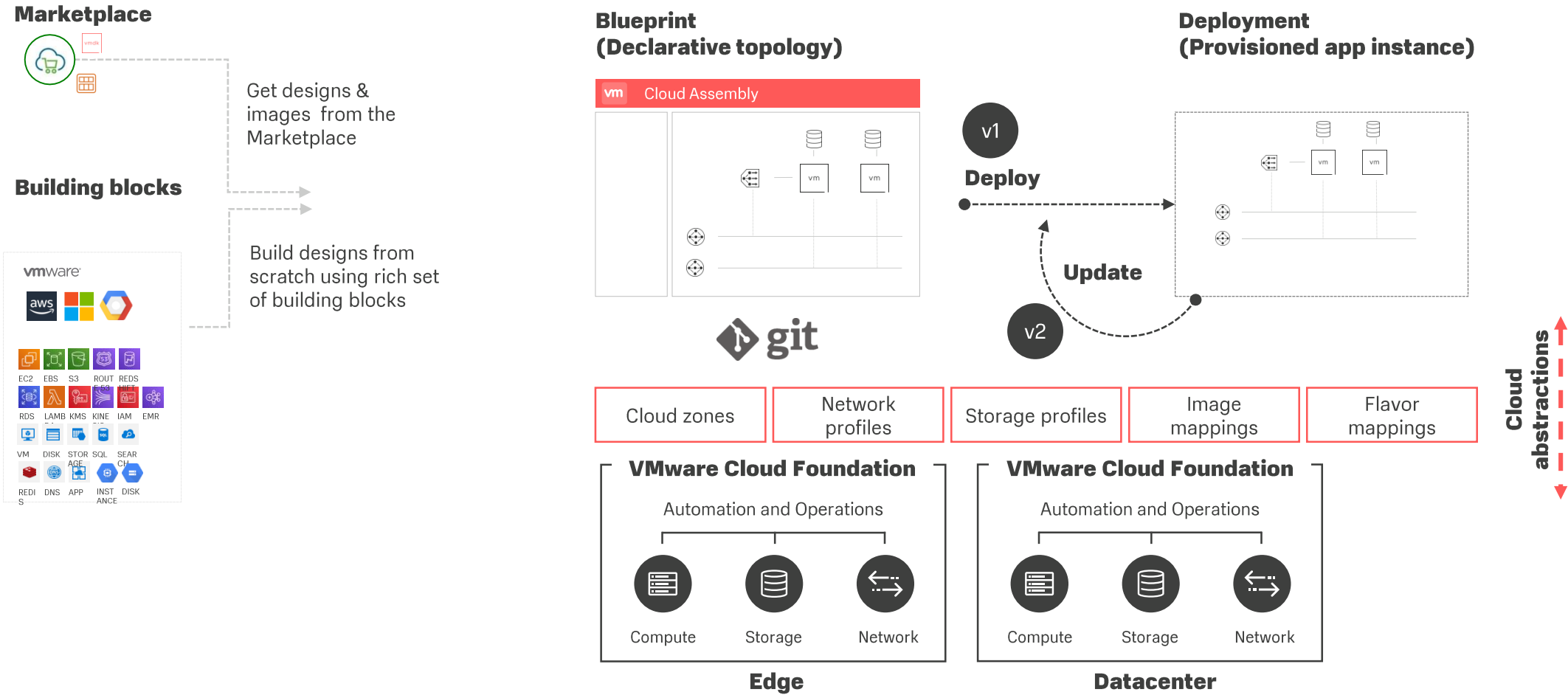
```
>>
1 formatVersion: 1
2 inputs: {}
3 resources:
4 Cloud_Machine_1:
5   type: Cloud.Machine
6   properties:
7   image: ''
8   flavor: ''
9   networks:
10  - name: '${resource.Cloud_NSX_Network_1.name}'
11 Cloud_NSX_Network_1:
12   type: Cloud.NSX.Network
13   properties:
14   name: ''
15   networkType: |
16
```

routed	schema
public	schema
private	schema
outbound	schema
existing	schema

**On-Demand [Provider] Networks**

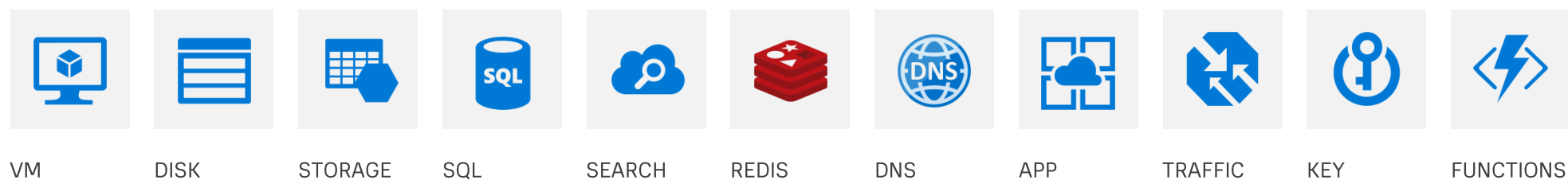
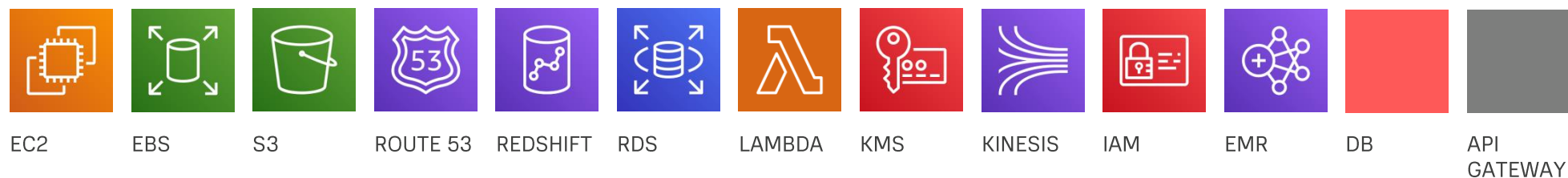
# Декларативный деплой и поддержка жизненного цикла

Последовательное обновление деплоя с помощью кода



# Использование облачных сервисов

Богатый выбор IaaS и PaaS



# Портал самообслуживания

Для запроса и автоматического предоставления ИТ-сервисов

The screenshot displays the 'Service Broker' portal interface. At the top, there is a navigation bar with 'Catalog', 'Deployments', and 'Administration' tabs. The main content area is titled 'Catalog Items' and shows a list of 17 items. The items are arranged in a grid and include:

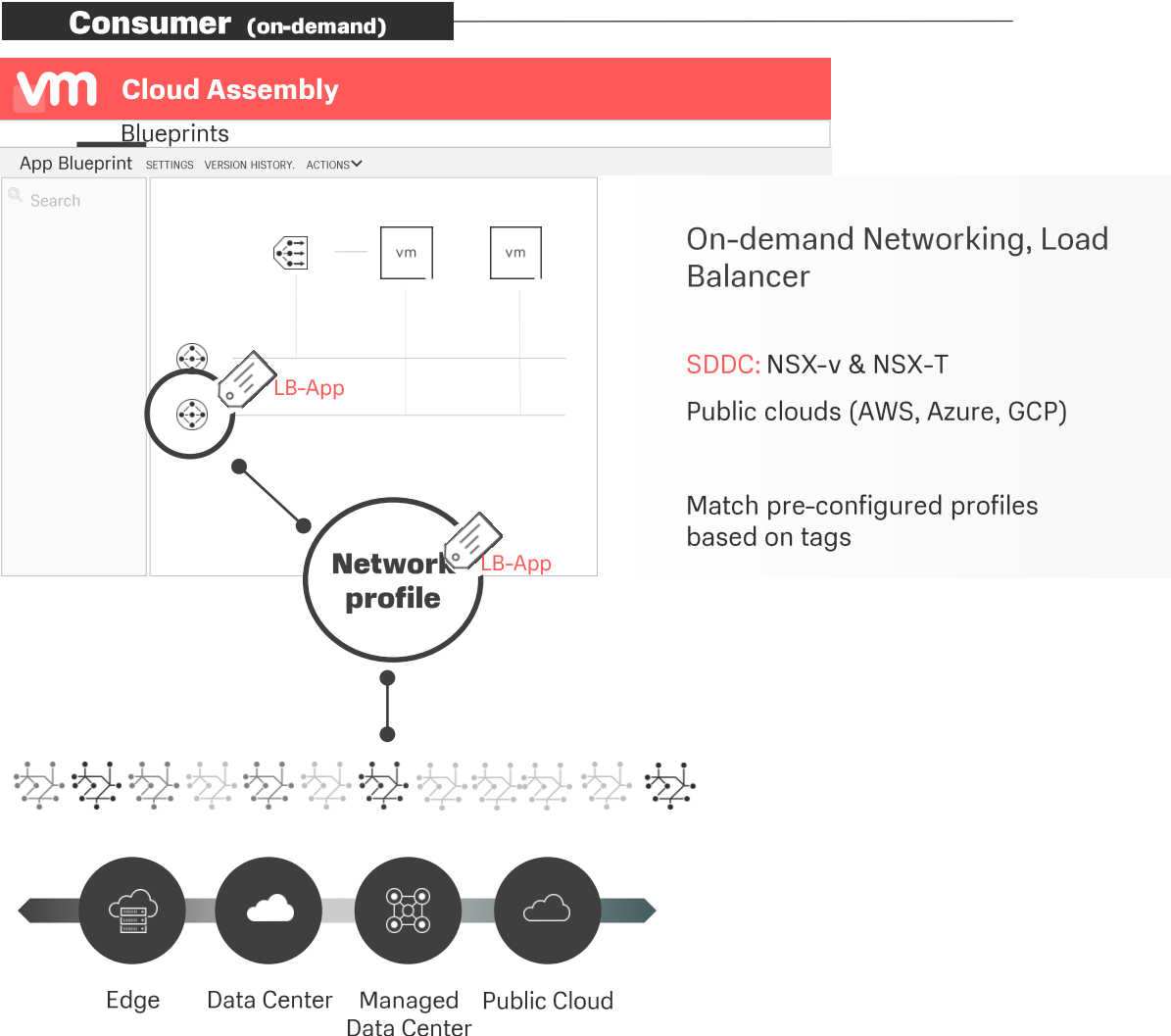
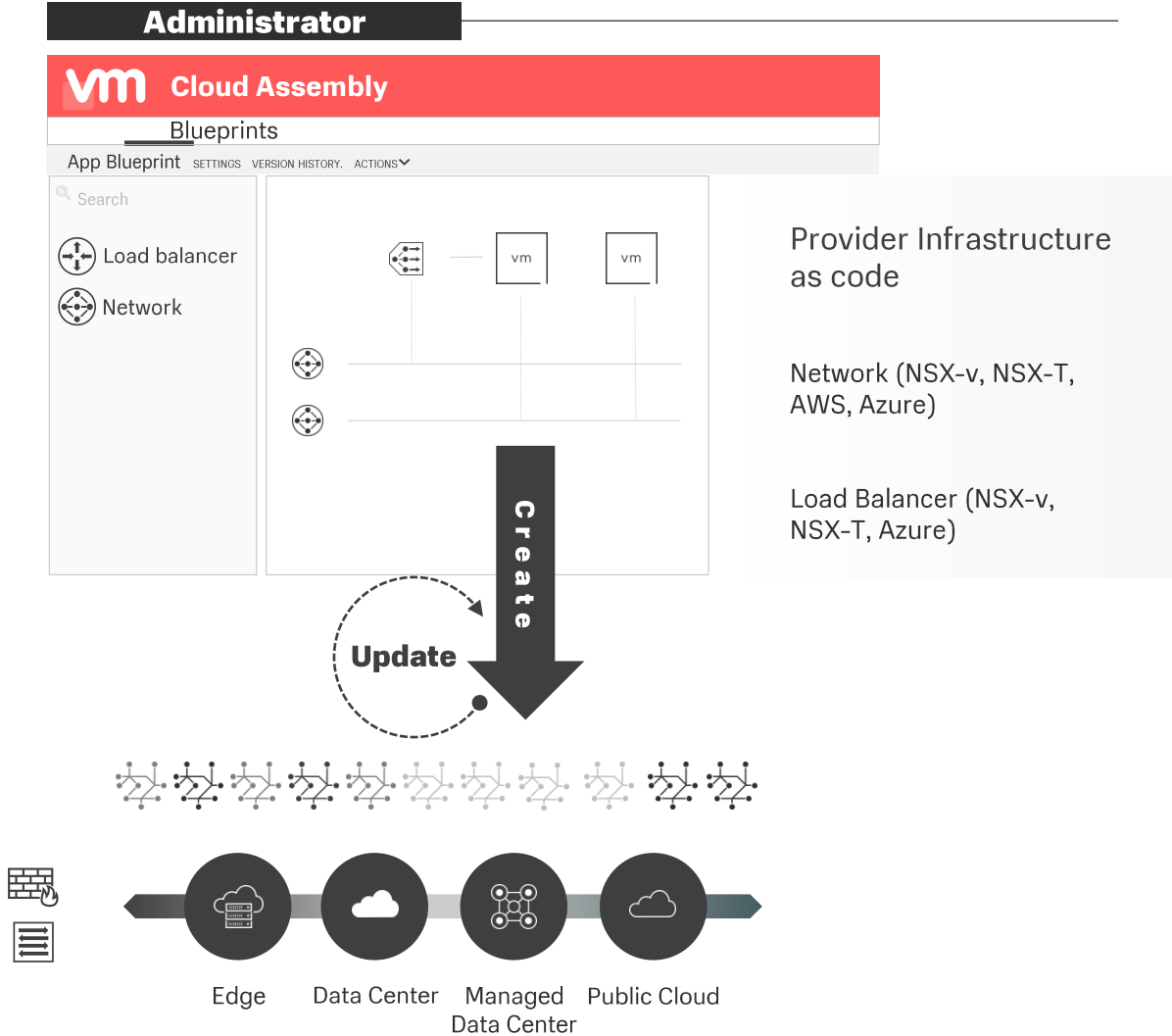
- Windows Server 2016 (MICROSOFT) - Windows Server 2016 Datacenter Edition - VMDK - REQUEST
- Redhat Enterprise Linux 7 (IBM) - Redhat Enterprise Linux 7 - VMDK - REQUEST
- Ubuntu Bionic Linux (CANONICAL) - Ubuntu Bionic Linux (18.04 LTS) - VMDK - REQUEST
- SUSE Linux Enterprise Server 15 - SUSE Linux Enterprise Server 15 - VMDK - REQUEST
- NSX Network (VMWARE) - Create a network in SDDC - CLOUD ASSEMBLY - REQUEST
- NSX Load balancer (VMWARE) - Redhat Enterprise Linux 7 - CLOUD ASSEMBLY - REQUEST
- NSX Security group (VMWARE) - Ubuntu Bionic Linux (18.04 LTS) - CLOUD ASSEMBLY - REQUEST
- Serverless app model (AWS) - SUSE Linux Enterprise Server 15 - AWS CFT - REQUEST

Портал самообслуживания для предоставления пользователям ИТ-сервисов.

Управление жизненным циклом сервисов: продление запрошенных сервисов, автоматическое удаление неиспользуемых сервисов.

# Кросс-облачная сеть и безопасность

Богатые возможности для админов. Простое использование для клиентов



# Поддержка провайдеров контейнерной инфраструктуры

Деплой, каталог, биллинг, дашборды



Управление жизненным  
циклом кластеров



Политики и управление



Самообслуживание  
для разработчиков



Управление приложениями

## Private cloud



PKS



WCP



OPENSIFT

## Public cloud



Cloud PKS



Amazon  
EKS



# Готовые сервисы для частного облака

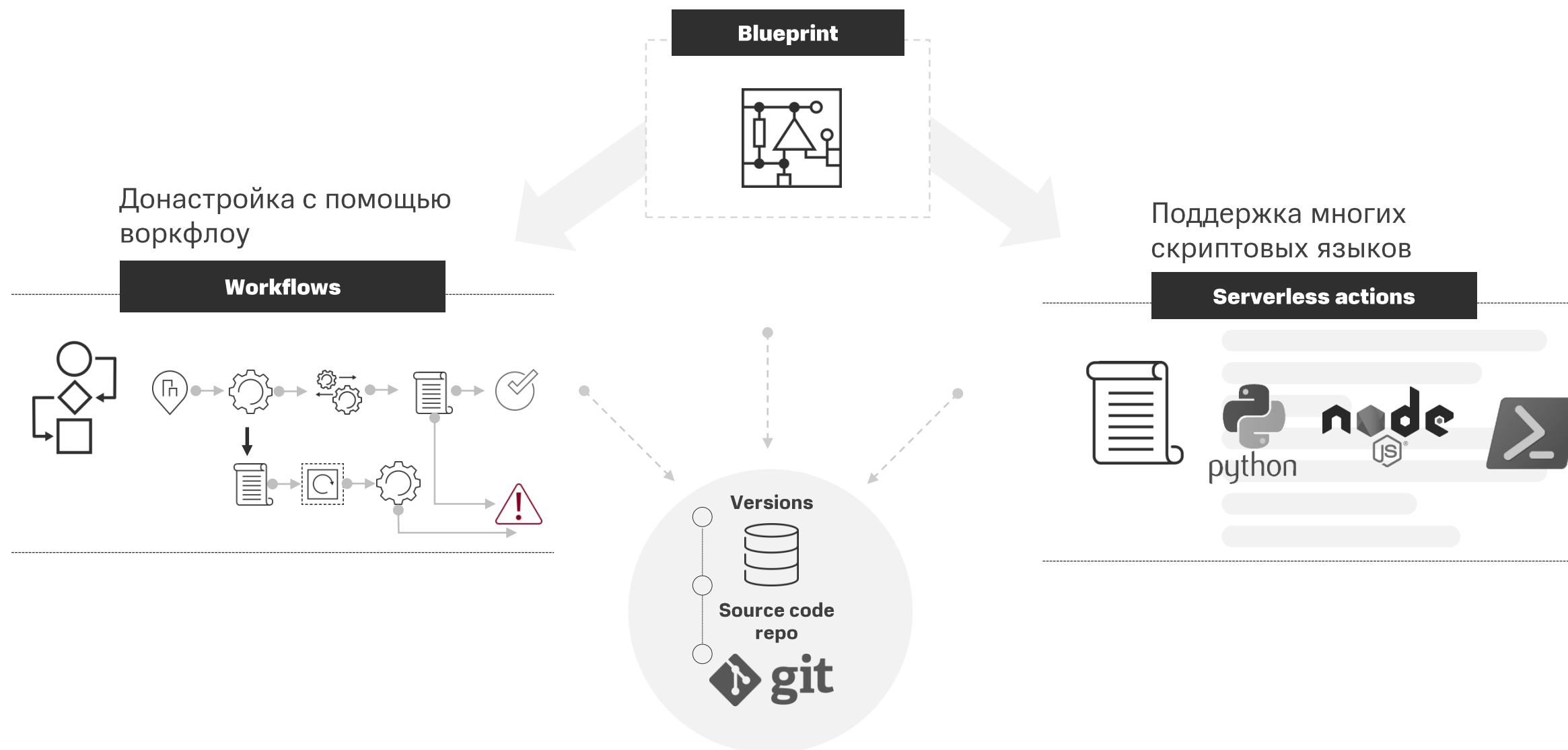
The screenshot displays the VMware Marketplace interface. At the top, there is a dark navigation bar with the VMware logo and 'VMware Marketplace™' on the left, and links for 'Blog', 'About', 'Sign-in', and 'Sign-up' on the right. The main content area features a grid of eight virtual appliance cards, each with a unique icon, a title, the creator 'By: Bitnami', a short description, and a set of three icons representing cloud, storage, and server resources. The appliances shown are:

- TYPO3 Virtual Appliance**: This TYPO3 virtual machine is secure, up-to-date, and pre-configured to work out of the box.
- Trac Virtual Appliance**: This Trac virtual machine is secure, up-to-date, and pre-configured to work out of the box.
- Tomcat Virtual Appliance**: This Tomcat virtual machine is secure, up-to-date, and pre-configured to work out of the box.
- Tiny Tiny RSS Virtual Appliance**: This Tiny Tiny RSS virtual machine is secure, up-to-date, and pre-configured to work out of the box.
- Tiki Wiki CMS Groupware Virtu...**: This Tiki Wiki CMS Groupware virtual machine is secure, up-to-date, and pre-configured to work out of the box.
- TestLink Virtual Appliance**: This TestLink virtual machine is secure, up-to-date, and pre-configured to work out of the box.
- TensorFlow Serving Virtual Ap...**: This TensorFlow Serving virtual machine is secure, up-to-date, and pre-configured to work out of the box.
- SuiteCRM Virtual Appliance**: This SuiteCRM virtual machine is secure, up-to-date, and pre-configured to work out of the box.

In the bottom right corner of the appliance grid, there is a small blue circular notification icon containing the number '2' and an information icon.

# Расширяемость на основе событий

Благодаря встроенной возможности оркестрации







**А как это  
работает?**

**Спасибо!**

