



DevCon School

Технологии будущего

# Использование Azure Resource Manager в управлении жизненным циклом приложений

Александр Шаповал  
Microsoft

## Что такое ARM?

---

Что представляет собой ARM API

Какова структура шаблонов ARM

Как использовать ARM-шаблоны для реализации Infrastructure as Code

## Как эффективно использовать ARM-ресурсы?

---

Как правильно выбрать шаблон VM

Как получить нужную производительность хранилища

Как обеспечить контроль доступа

## Как использовать ARM локально?

---

Что представляет собой Azure Stack

Как использовать ARM для локальных и гибридных решений

# Azure Resource Manager и группы ресурсов

#msdevcon

# Azure Service Management (ASM)

- Azure Service Management – первая версия API, которая предоставляет программный доступ к возможностям Azure
  - Предоставляется в виде REST API
- ASM основан на XML, для создания и запуска скриптов могут использоваться PowerShell, CLI
- ASM может использоваться для настройки таких ресурсов как: Cloud Services, Storage accounts, Virtual Networks
- Пакеты облачных приложений требуют наличия сертификатов, которые отделены от кода и загружаются с помощью портала управления

# ASM: недостатки и ограничения

- Невозможно с помощью одного скрипта задействовать несколько регионов или несколько сервисов
  - Для подобных сценариев необходимо использовать несколько скриптов и инструменты оркестрации
- Для реализации концепции «пространственной близости» используются территориальные группы (affinity groups)
- Отсутствует согласованность в предоставляемых сервисах API
  - Некоторые используют XML, некоторые JSON

# ASM: недостатки и ограничения

- Сложно управлять большим количеством ресурсов в рамках организации
- Ограниченные средства контроля доступа
  - Доп. администратор подписки для привилегированных действий
- Ограниченные средства аудита, доступные на портале
  - **List Subscription Operations** возвращает список операций **create**, **update** и **delete**, выполненных в рамках подписки за определенный интервал времени

# Azure Resource Manager (ARM)

Уровень  
управления

Инструменты



Microsoft Azure

+



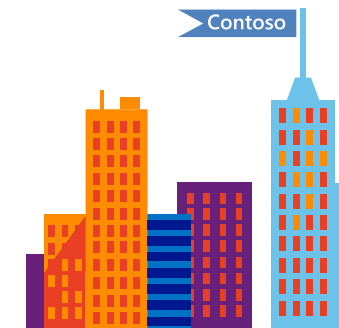
Command line

+



Visual Studio

Расширения



Провай-  
деры





# Направления усовершенствований



Развертывание



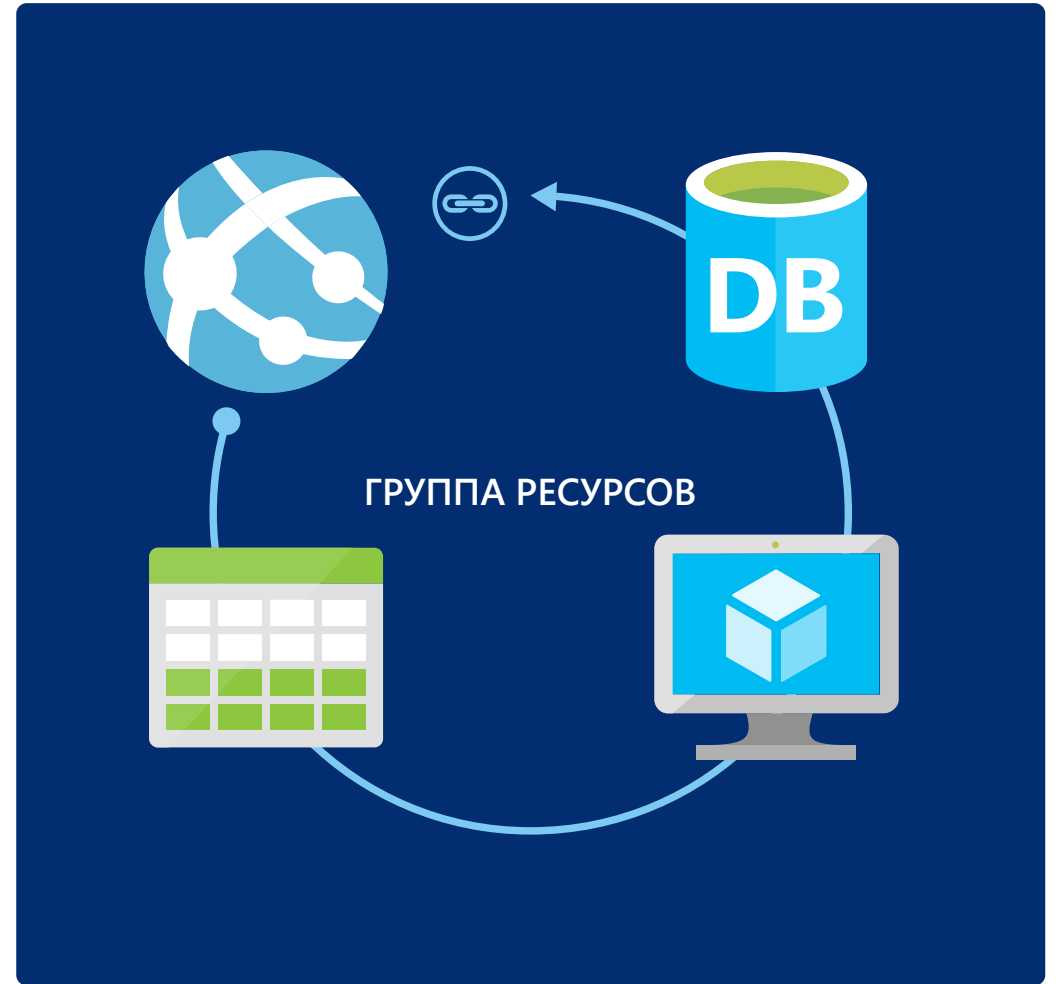
Структурирование



Контроль

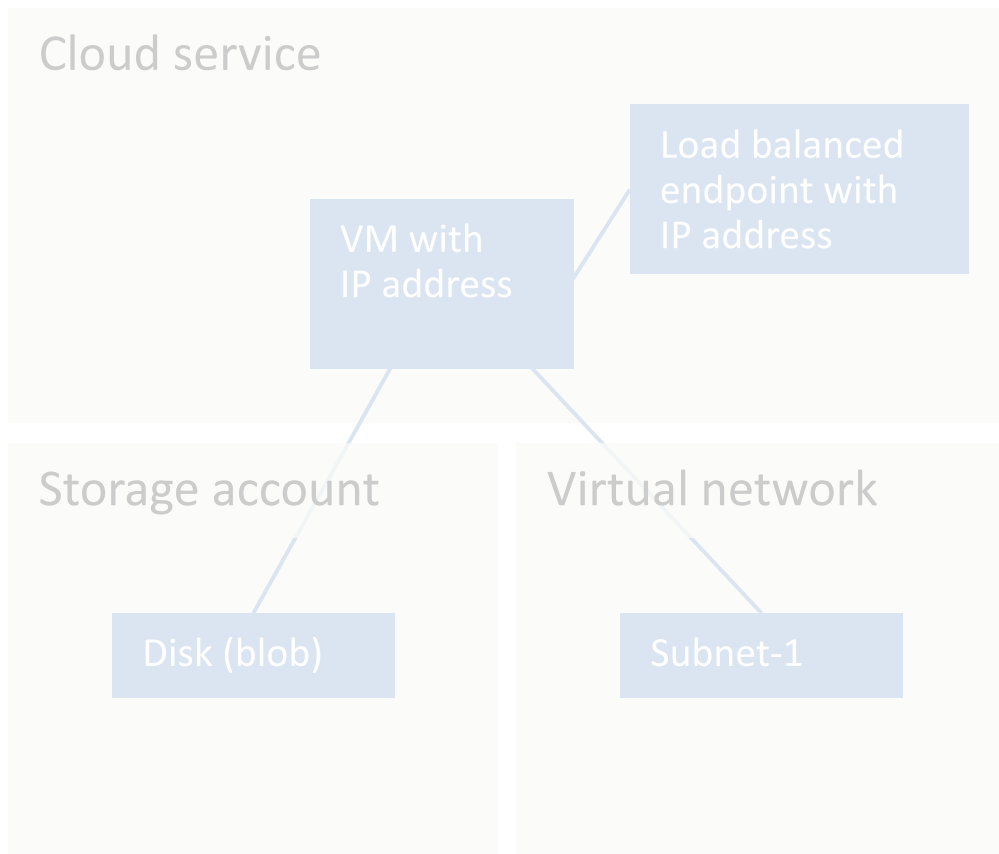
# Группы ресурсов

- Контейнеры с множеством экземпляров ресурсов
- Каждый экземпляр относится к определенному типу ресурса
- Типы ресурсов определяются провайдерами ресурсов
- Каждый ресурс *должен* принадлежать одной и только одной группе ресурсов

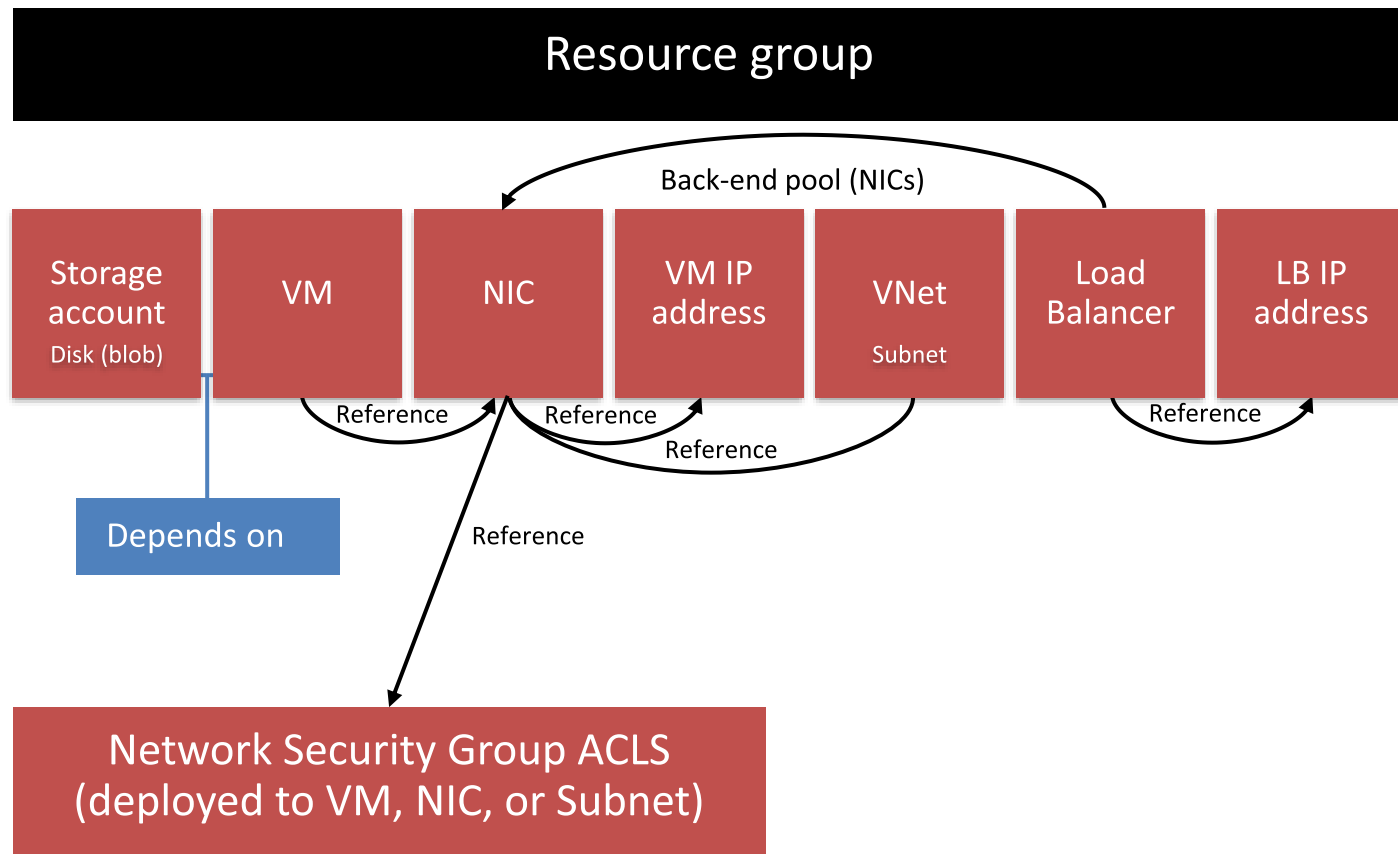


# Пример использования Resource Manager

## Классическая модель (v1)



## Resource Manager (v2)



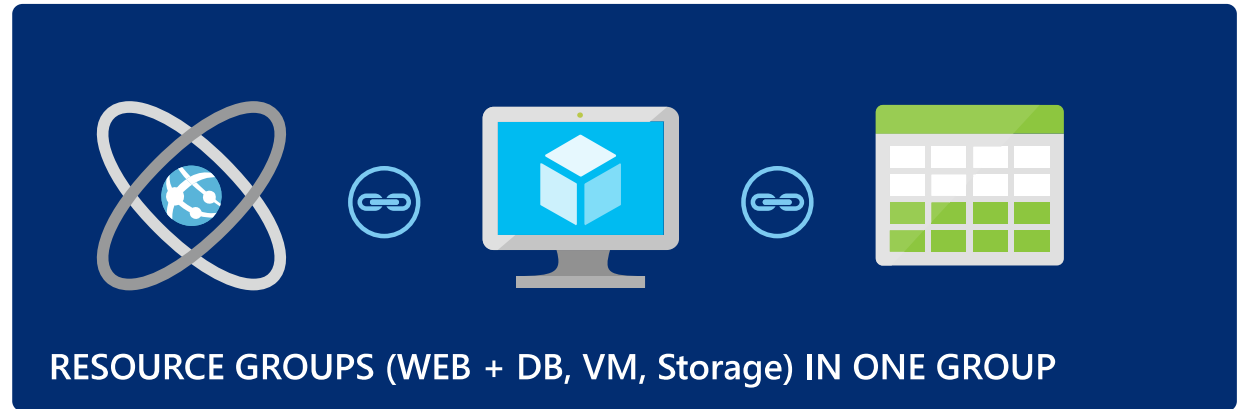
# Жизненный цикл группы ресурсов

Вопрос:

Должны некоторые ресурсы принадлежать одной группе или разным?

Ответ:

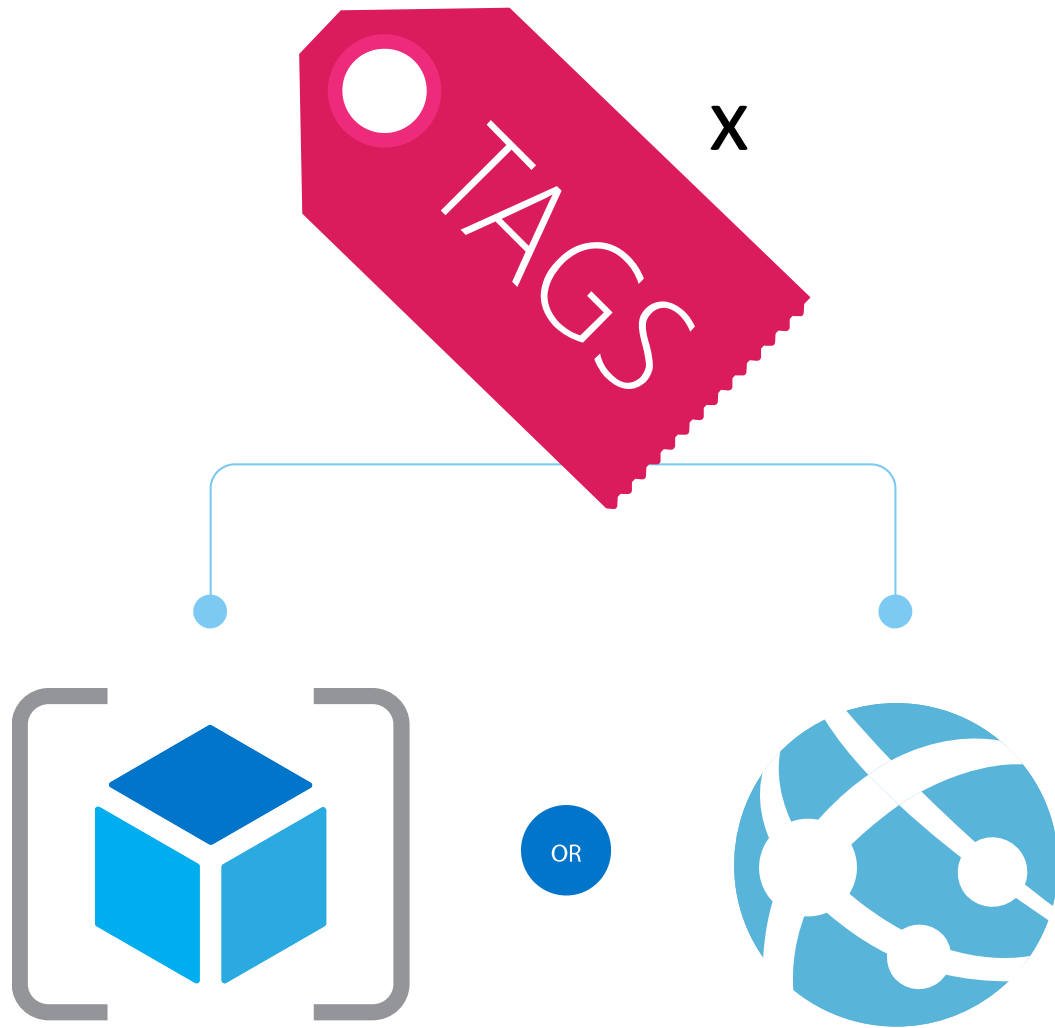
Определяется тем, имеют ли они общий жизненный цикл и общее управление



OR



# Теги ресурсов



- Пара «имя-значение», присвоенная ресурсам или группам ресурсов
- Могут применяться в рамках подписки
- Каждый ресурс может иметь до 15 тегов
- Различные принципы тегирования, такие как: принадлежность к роли, отделу, окружению и пр.

 Демонстрация

# Создание виртуальной машины на основе ARM с помощью портала

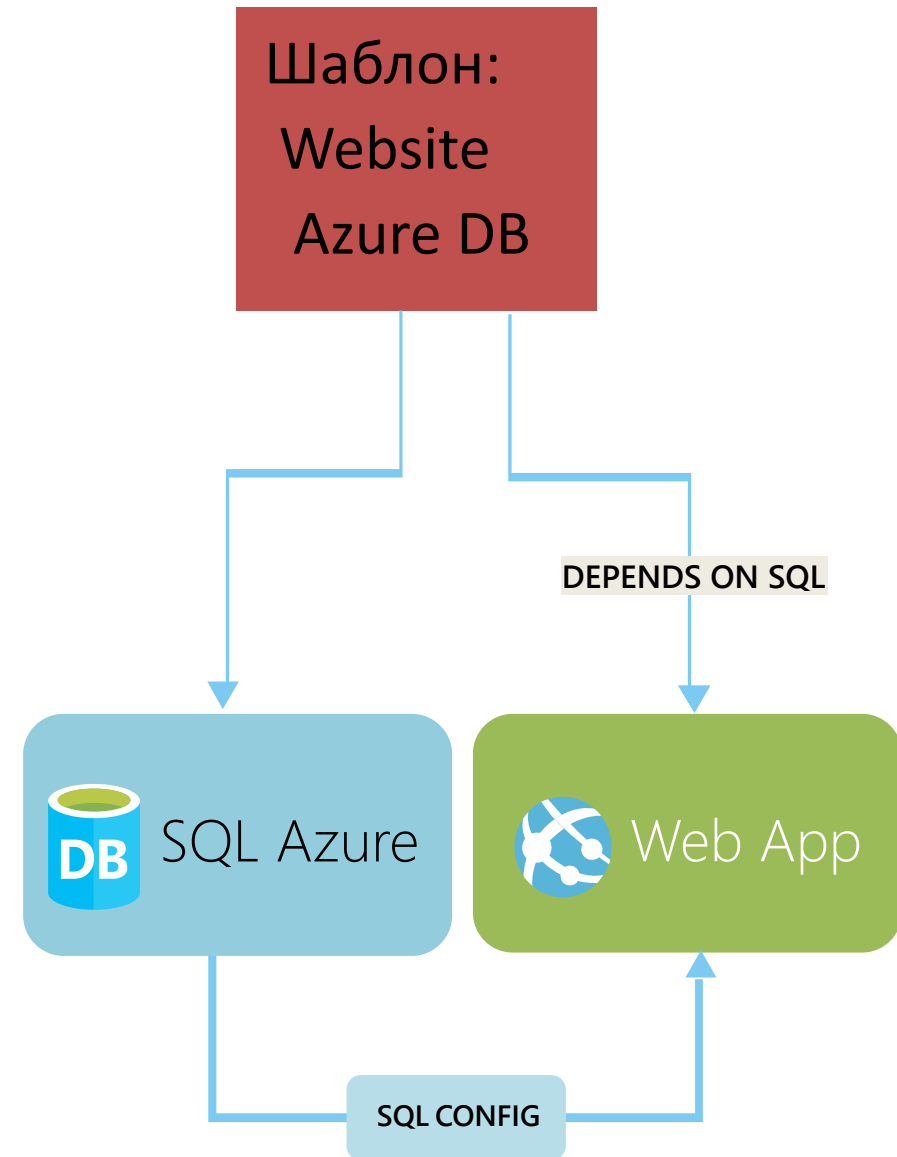
#msdevcon

# Шаблоны Azure Resource Manager

#msdevcon

# Шаблоны ресурсов

- Основанная на модели декларативная спецификация ресурсов, их конфигурации, кода, расширений
- Многократная применимость
- Согласованное развертывание
- Использование в системах контроля версий
- Параметризация ввода/вывода





# Структура шаблона

```
{
  "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "",
  "parameters": { },
  "variables": { },
  "resources": [ ],
  "outputs": { }
}
```

ELEMENT NAME	REQUIRED	DESCRIPTION
\$schema	Yes	Location of the JSON schema file that describes the version of the template language.
contentVersion	Yes	Version of the template (such as 1.0.0.0). When deploying resources using the template, this value can be used to make sure that the right template is being used.
parameters	No	Values that are provided when deployment is executed to customize resource deployment.
variables	No	Values that are used as JSON fragments in the template to simplify template language expressions.
resources	Yes	Types of services that are deployed or updated in a resource group.
outputs	No	Values that are returned after deployment.

# Параметры

```
"parameters": {  
  "<parameterName>" : {  
    "type" : "<type-of-parameter-value>",  
    "defaultValue": "<optional-default-value-of-parameter>",  
    "allowedValues": [ "<optional-array-of-allowed-values>" ]  
  }  
}
```

ELEMENT NAME	REQUIRED	DESCRIPTION
parameterName	Yes	Name of the parameter. Must be a valid JavaScript identifier.
type	Yes	Type of the parameter value. See the list below of allowed types.
defaultValue	No	Default value for the parameter, if no value is provided for the parameter.
allowedValues	No	Array of allowed values for the parameter to make sure that the right value is provided.

The allowed types and values are:

- string or secureString - any valid JSON string
- int - any valid JSON integer
- bool - any valid JSON boolean
- object or secureObject - any valid JSON object
- array - any valid JSON array

# Пример параметров

```
"parameters": {  
  "siteName": {  
    "type": "string"  
  },  
  "siteLocation": {  
    "type": "string"  
  },  
  "hostingPlanName": {  
    "type": "string"  
  },  
  "hostingPlanSku": {  
    "type": "string",  
    "allowedValues": [  
      "Free",  
      "Shared",  
      "Basic",  
      "Standard",  
      "Premium"  
    ],  
    "defaultValue": "Free"  
  }  
}
```

# Переменные

```
"parameters": {
  "username": {
    "type": "string"
  },
  "password": {
    "type": "secureString"
  }
},
"variables": {
  "connectionString": "[concat('Name=', parameters('username'), ';Password=', parameters('password'))]"
}
```

```
"parameters": {
  "environmentName": {
    "type": "string",
    "allowedValues": [
      "test",
      "prod"
    ]
  }
},
"variables": {
  "environmentSettings": {
    "test": {
      "instancesSize": "Small",
      "instancesCount": 1
    },
    "prod": {
      "instancesSize": "Large",
      "instancesCount": 4
    }
  },
  "currentEnvironmentSettings": "[variables('environmentSettings')[parameters('environmentName')]]",
  "instancesSize": "[variables('currentEnvironmentSettings').instancesSize]",
  "instancesCount": "[variables('currentEnvironmentSettings').instancesCount]"
}
```

# Ресурсы

```
"resources": [
  {
    "apiVersion": "<api-version-of-resource>",
    "type": "<resource-provider-namespace/resource-type-name>",
    "name": "<name-of-the-resource>",
    "location": "<location-of-resource>",
    "tags": "<name-value-pairs-for-resource-tagging>",
    "dependsOn": [
      "<array-of-related-resource-names>"
    ],
    "properties": "<settings-for-the-resource>",
    "resources": [
      "<array-of-dependent-resources>"
    ]
  }
]
```

ELEMENT NAME	REQUIRED	DESCRIPTION
apiVersion	Yes	Version of the API that supports the resource. For the available versions and schemas for resources, see <a href="#">Azure Resource Manager Schemas</a> .
type	Yes	Type of the resource. This value is a combination of the namespace of the resource provider and the resource type that the resource provider supports.
name	Yes	Name of the resource. The name must follow URI component restrictions defined in RFC3986.
location	No	Supported geo-locations of the provided resource.
tags	No	Tags that are associated with the resource.
dependsOn	No	Resources that the resource being defined depends on. The dependencies between resources are evaluated and resources are deployed in their dependent order. When resources are not dependent on each other, they are attempted to be deployed in parallel. The value can be a comma separated list of a resource names or resource unique identifiers.
properties	No	Resource specific configuration settings.
resources	No	Child resources that depend on the resource being defined.

# Пример ресурса

```
"resources": [  
  {  
    "apiVersion": "2014-06-01",  
    "type": "Microsoft.Web/serverfarms",  
    "name": "[parameters('hostingPlanName')]",  
    "location": "[resourceGroup().location]",  
    "properties": {  
      "name": "[parameters('hostingPlanName')]",  
      "sku": "[parameters('hostingPlanSku')]",  
      "workerSize": "0",  
      "numberOfWorkers": 1  
    }  
  },  
  {  
    "apiVersion": "2014-06-01",  
    "type": "Microsoft.Web/sites",  
    "name": "[parameters('siteName')]",  
    "location": "[resourceGroup().location]",  
    "tags": {  
      "environment": "test",  
      "team": "ARM"  
    },  
    "dependsOn": [  
      "[resourceId('Microsoft.Web/serverfarms', parameters('hostingPlanName'))]"  
    ],  
    "properties": {  
      "name": "[parameters('siteName')]",  
      "serverFarm": "[parameters('hostingPlanName')]"  
    },  
    "resources": [  
      {  
        "apiVersion": "2014-06-01",  
        "type": "Extensions",  
        "name": "MSDeploy",  
        "properties": {  
          "packageUri": "https://auxmktplceprod.blob.core.windows.net/packages/StarterSite-modified.zip",  
          "dbType": "None",  
          "connectionString": "",  
          "setParameters": {  
            "Application Path": "[parameters('siteName')]"  
          }  
        }  
      }  
    ]  
  }  
]
```

# Выходные данные (Outputs)

```
"outputs": {  
  "<outputName>" : {  
    "type" : "<type-of-output-value>",  
    "value": "<output-value-expression>",  
  }  
}
```

ELEMENT NAME	REQUIRED	DESCRIPTION
outputName	Yes	Name of the output value. Must be a valid JavaScript identifier.
type	Yes	Type of the output value. Output values support the same types as template input parameters.
value	Yes	Template language expression which will be evaluated and returned as output value.

```
"outputs": {  
  "siteUri" : {  
    "type" : "string",  
    "value": "[concat('http://',reference(resourceId('Microsoft.Web/sites', parameters('siteName'))).hostNames[0])]"  
  }  
}
```

# Выходные данные (Outputs)

- Шаблон может возвращать значения с помощью секции outputs

```
"outputs": {  
  "masterip": {  
    "value":  
      "[reference(concat(variables('nicName'),0)).ipConfigurations[0].properties.privateIPAddress]",  
    "type": "string"  
  }  
}
```

- Эти значения может использовать вызывающий

```
"masterIpAddress": {  
  "value":  
    "[reference('master-node').outputs.masterip.value]"  
}
```



# Функции и выражения языка шаблона

`copyIndex(offset)` – возвращает текущий индекс итерации

`length(array)` – возвращает число элементов в массиве

`listKeys(storageAccountResourceId, apiVersion)` – возвращает ключи учетной записи хранения

`parameters('parameterName')` – возвращает значение параметра

`providers(namespace, resourceType)` – возвращает информацию о провайдере ресурса

`resourceGroup()` – возвращает структурированный объект, который представляет собой текущую группу ресурсов

`resourceId('namespace/resourceType', 'resourceName')` – возвращает уникальный идентификатор ресурса при ссылке на объект на пределах текущей группы ресурсов

`subscription()` – возвращает информацию о подписке

`variables('variables')` – возвращает значение переменной

# Несколько экземпляров ресурсов

```
"parameters": {  
  "count": {  
    "type": "int",  
    "defaultValue": 3  
  }  
},  
"resources": [  
  {  
    "name": "[concat('examplecopy-', copyIndex())]",  
    "type": "Microsoft.Web/sites",  
    "location": "East US",  
    "apiVersion": "2014-06-01",  
    "copy": {  
      "name": "websitescopy",  
      "count": "[parameters('count')]"  
    },  
    "properties": {}  
  }  
]
```

- examplecopy-0
- examplecopy-1
- examplecopy-2.

# Определение зависимостей

```
{  
  "name": "<name-of-the-resource>",  
  "type": "<resource-provider-namespace/resource-type-name>",  
  "apiVersion": "<supported-api-version-of-resource>",  
  "location": "<location-of-resource>",  
  "tags": { <name-value-pairs-for-resource-tagging> },  
  "dependsOn": [ <array-of-related-resource-names> ],  
  "properties": { <settings-for-the-resource> },  
  "resources": { <dependent-resources> }  
}
```

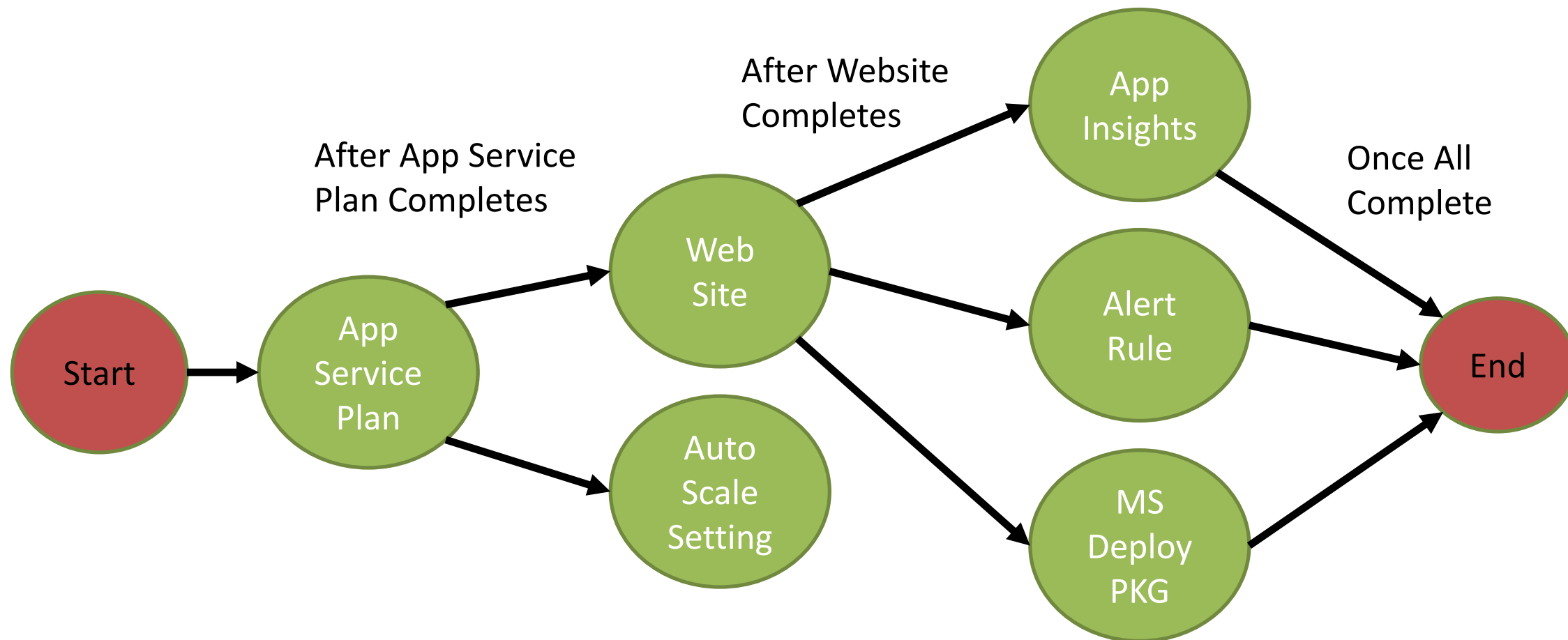
**dependsOn:** определяет зависимости от других ресурсов, которые должны быть доступны перед созданием данного ресурса. Влияет на скорость развертывания

**resources:** указывает на дочерние ресурсы, которые не влияют на процесс оптимизации развертывания. Можно определить до 5 уровней дочерних ресурсов в глубину, при этом функциональность **dependsOn** не обеспечивается

**reference:** функция определяет неявную зависимость и влияет на порядок развертывания ресурсов. Не используйте одновременно **dependsOn** и **reference** для определения зависимостей конкретного ресурса, достаточно какого-то одного варианта. Использование **reference** предпочтительнее, так как позволяет минимизировать количество последовательных шагов развертывания

# Реализация шаблона

- Модуль выполнения строит машину состояния
- `dependsOn()` и `reference()` определяют зависимости



 Демонстрация

# Создание виртуальной машины на основе ARM с помощью шаблона

#msdevcon

 Демонстрация

# Разработка ARM-шаблонов

#msdevcon

# Проектирование инфраструктуры Azure для высокопроизводительных вычислений и хранения данных

# Основные критерии

- Семейство VM
- Размер VM
- Производительность дисковой подсистемы



# Семейства виртуальных машин

- A: A0-A7 и A8-A11
- D, Dv2, DS, DSv2
- F, Fs
- G, GS
- H
- N (preview)

# Аппаратный кластер (hardware cluster)

## Определение

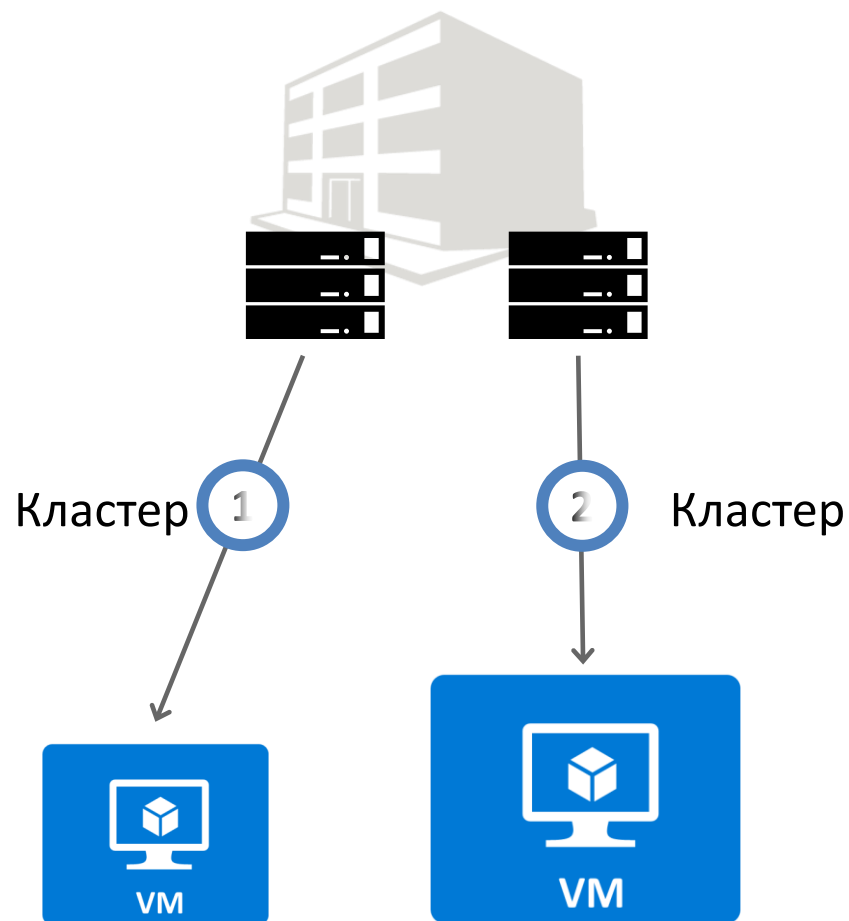
Аппаратный кластер осуществляет поддержку виртуальных машин определенных размеров

Каждая **облачная служба** привязана к одному аппаратному кластеру

Каждая **территориальная группа** с одной или более VM привязана к одному аппаратному кластеру

## Влияние размера

Размер VM может изменяться только в рамках поддерживаемого кластером диапазона



# Azure Storage

Для хранения данных в Azure используются учетные записи хранения (storage accounts)

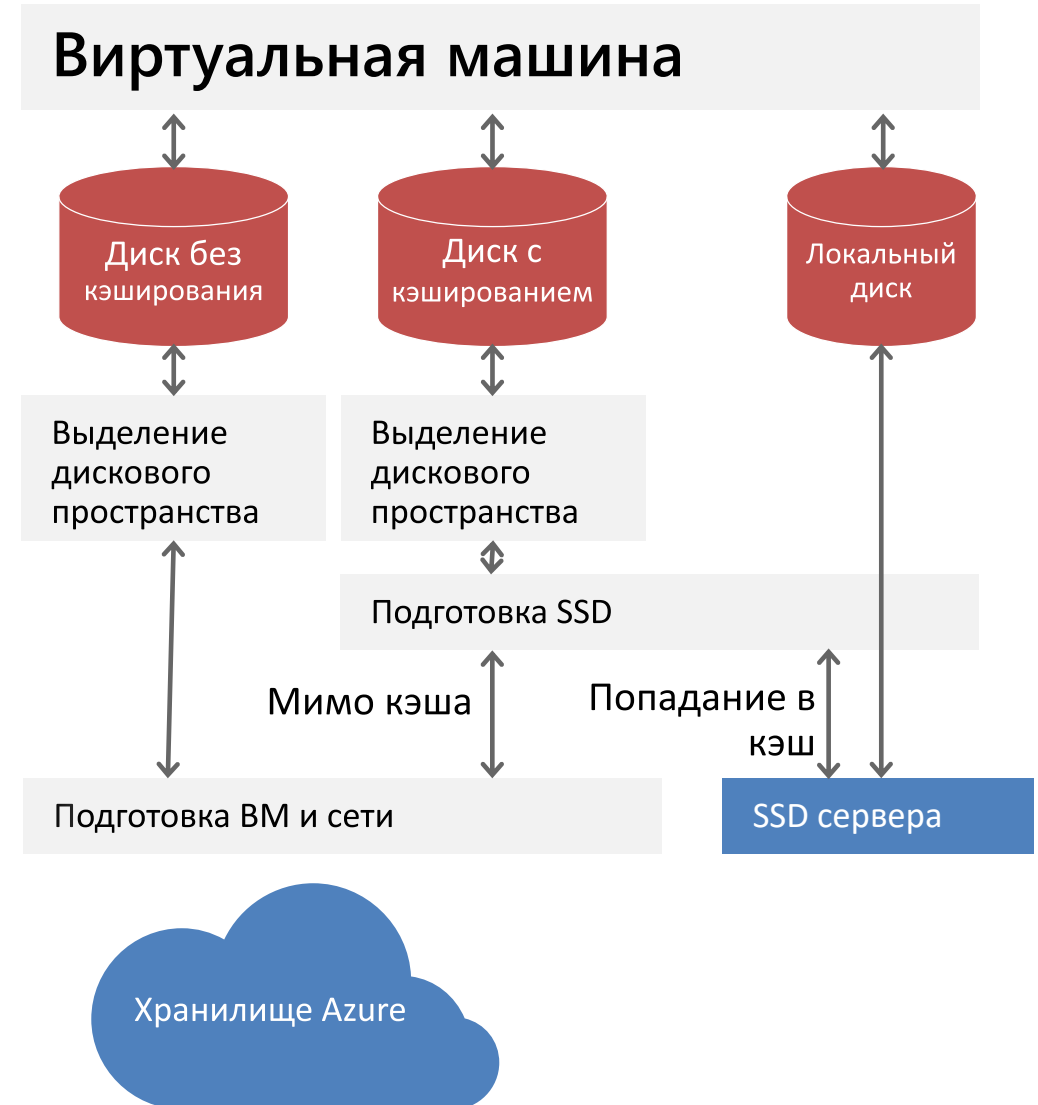
Хранилище может быть двух типов: стандартное (standard) и премиальное (premium)

Для любого типа хранилища Azure применяет лимиты:

Max IOPS на storage account (20000)

Max IOPS на виртуальный жесткий диск

Для премиального хранилища используются дополнительные параметры и лимиты



# Premium Storage

- Высокая пропускная способность и низкая задержка
- Емкость хранилища (Premium storage account) до 35 ТБ
- До 80 000 операций ввода-вывода в секунду для VM
- До 5000 операций ввода-вывода в секунду для диска
- Около 5 мс на операции чтения и записи (без кэша)
- Задержка при операции чтения менее 1 мс (кэш)

# Анализ параметров производительности и существующих ограничений

Существуют лимиты на количество IOPS и на пропускную способность диска

Лимиты установлены на диск, на VM и на учетную запись хранения

Не более 20 000 IOPS на учетную запись хранения (premium storage account)

Тип диска хранения	P10	P20	P30
Размер диска	128 ГБ	512 ГБ	1024 ГБ (1 ТБ)
Количество операций ввода-вывода в секунду для каждого диска	500	2300	5000
Пропускная способность диска	100 МБ в секунду	150 МБ в секунду	200 МБ в секунду

Размер VM	Число ядер ЦП	Макс. число операций ввода-вывода в секунду для диска (на одну VM)	Максимальная пропускная способность диска (на одну VM)	Размер кэша (ГБ)
STANDARD_DS1	1	3200	32 МБ в секунду	43
STANDARD_DS2	2	6400	64 МБ в секунду	86
STANDARD_DS3	4	12 800	128 МБ в секунду	172
STANDARD_DS4	8	25 600	256 МБ в секунду	344
STANDARD_DS11	2	6400	64 МБ в секунду	72
STANDARD_DS12	4	12 800	128 МБ в секунду	144
STANDARD_DS13	8	25 600	256 МБ в секунду	288
STANDARD_DS14	16	50 000	512 МБ в секунду	576
STANDARD_GS1	2	5000	125 МБ в секунду	264
STANDARD_GS2	4	10 000	250 МБ в секунду	528
STANDARD_GS3	8	20 000	500 МБ в секунду	1056
STANDARD_GS4	16	40 000	1000 МБ в секунду	2112
STANDARD_GS5	32	80 000	2000 МБ в секунду	4224

# Сколько учетных записей хранения требуется?

Это зависит от ограничений (диск или VM)...



или



## ПРИМЕР:

5 VM на дисках P30 (макс. 5000 операций ввода-вывода в секунду) **или**

Одна VM с пятью чередующимися дисками P30 = 25 000 операций ввода-вывода в секунду

Значит, в обоих случаях понадобятся две учетные записи хранения, чтобы достичь уровня 25 000 операций ввода-вывода в секунду



=



12 000 IOPS

## ПРИМЕР:

VM поддерживают до 12 000 операций ввода-вывода в секунду **Нам нужно три таких машины...**

$(3 \times 12\,000 = 36\,000) : 20\,000 = 2$  учетные записи хранения



=



**Максимум: 35 ТБ**

## ПРИМЕР:

Максимальное допустимое дисковое пространство для учетных записей премиум-класса – 35 ТБ

Чтобы получить 64 дисков по 1 ТБ (это допустимо для VM GS5), понадобятся две учетные записи хранения

# Работа с временным диском

**Никогда не размещайте критически важные непродублированные данные на временном диске!**

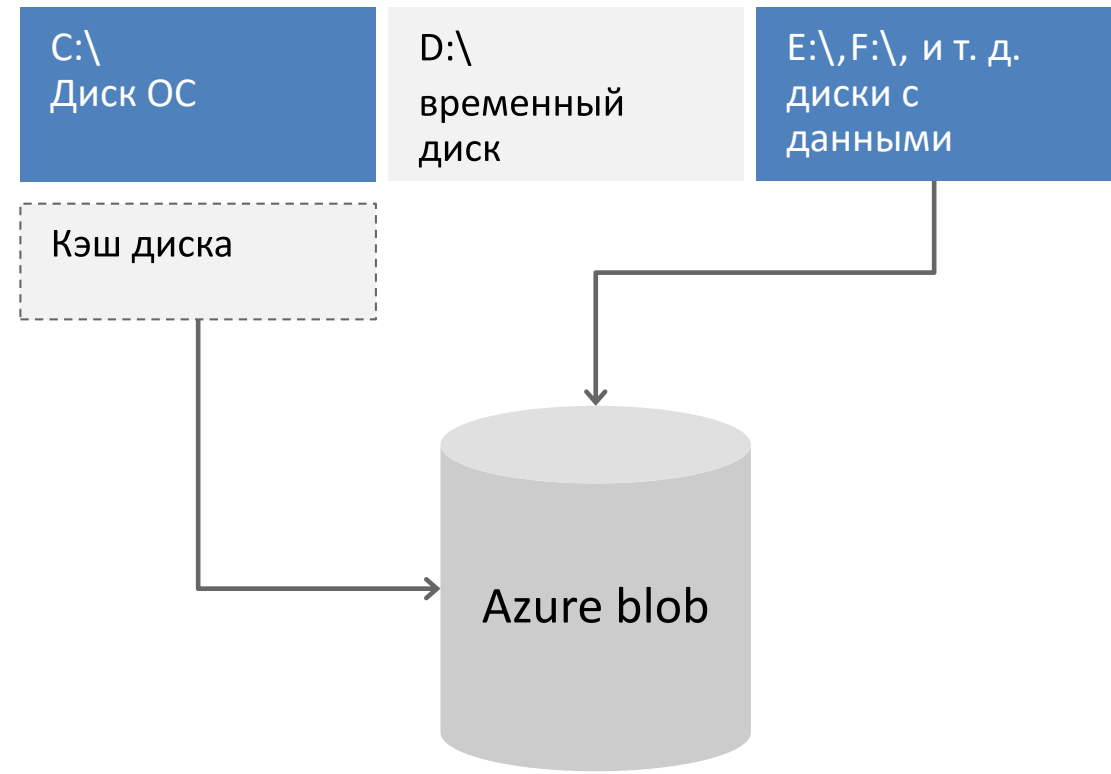
**Используйте его только для работы с SQL TempDB и Buffer Pool Extensions на VM серий D и G (временные диски SSD)**

<http://blogs.technet.com/b/dataplatforminsider/archive/2014/09/25/using-ssds-in-azure-vms-to-store-sql-server-tempdb-and-buffer-pool-extensions.aspx>

**Используйте планировщик для задач на временных дисках**

**Тестируйте запланированные задачи при помощи операции по изменению размера VM**

## Виртуальная машина Azure



 Демонстрация

# Сравнение производительности стандартного и премиального хранилищ

#msdevcon



# Проектирование сетевой инфраструктуры Azure для повышения безопасности

#msdevcon

# Виртуальная сеть Azure

Стройте собственные сети

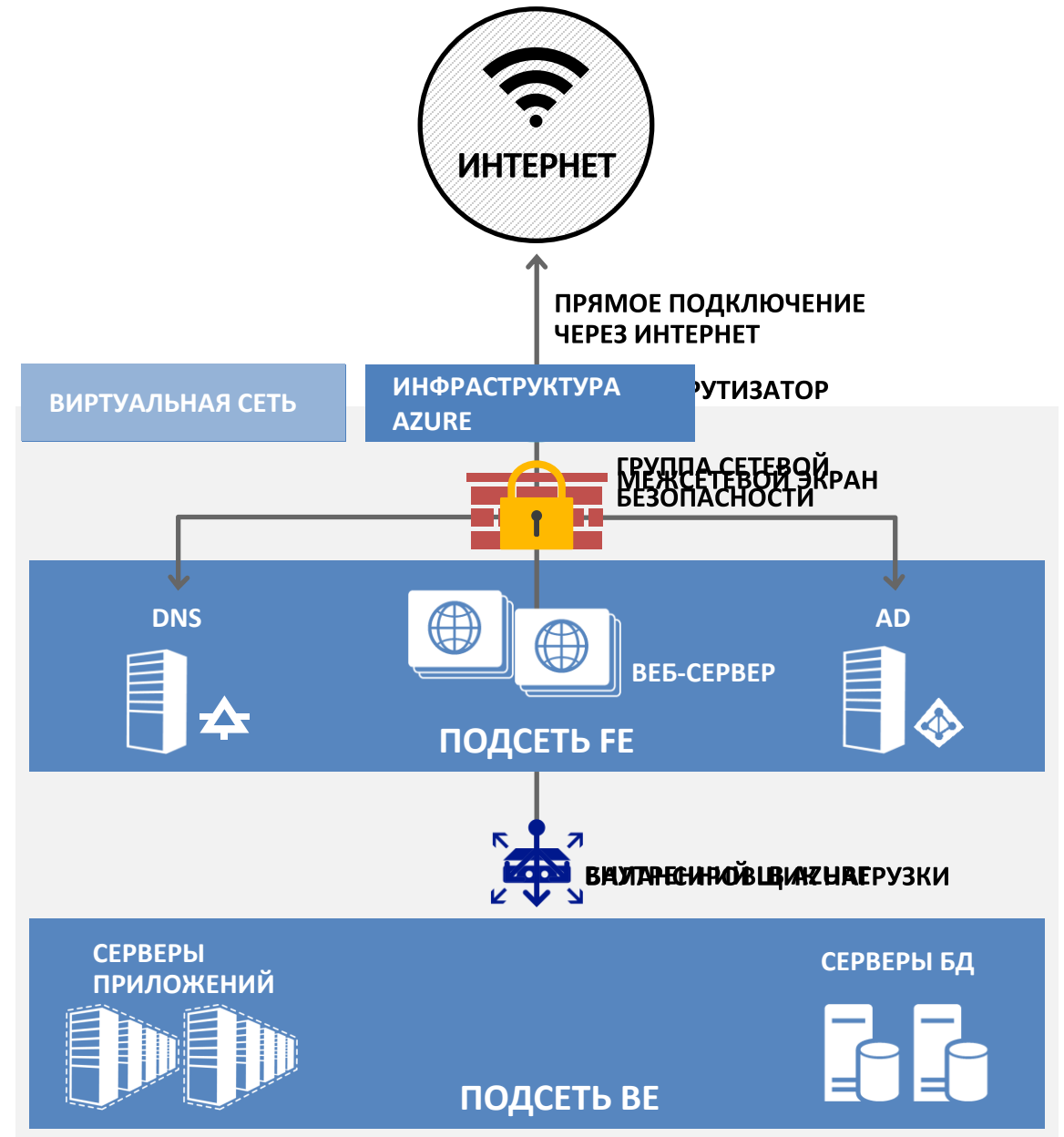
Логическая изоляция с контролем сетевых операций

Создавайте подсети с частными или общедоступными IP-адресами

Используйте собственную DNS или DNS Azure

Защитите VM с помощью групп сетевой безопасности

Запускайте высокодоступные внутренние службы с подсистемой балансировки нагрузки



# Адресация в классической модели

## VIP – Virtual IP address

- Публичный IP, не привязан к конкретной VM или сетевому адаптеру.
- Присваивается облачной службе.
- Облачная служба может включать в себя несколько VM, которые, таким образом, разделяют VIP.

## DIP – Dynamic IP address

- Динамически (с помощью DHCP) присваивается VM. Не меняйте этот адрес вручную!
- Срок аренды равен сроку жизни VM.
- При создании в виртуальной сети VM получает DIP из диапазона этой сети.

# Адресация в классической модели

CLOUD SERVICE

VIP- 137.135.64.110



# IP-адреса и балансировка нагрузки в ARM

## Публичные IP-адреса в Azure

Присваиваются VM, балансировщикам, VPN-шлюзам, шлюзам приложений

## Public IP для VM

IP-адрес, эксклюзивно выделенный одной VM

Весь диапазон портов доступен по умолчанию

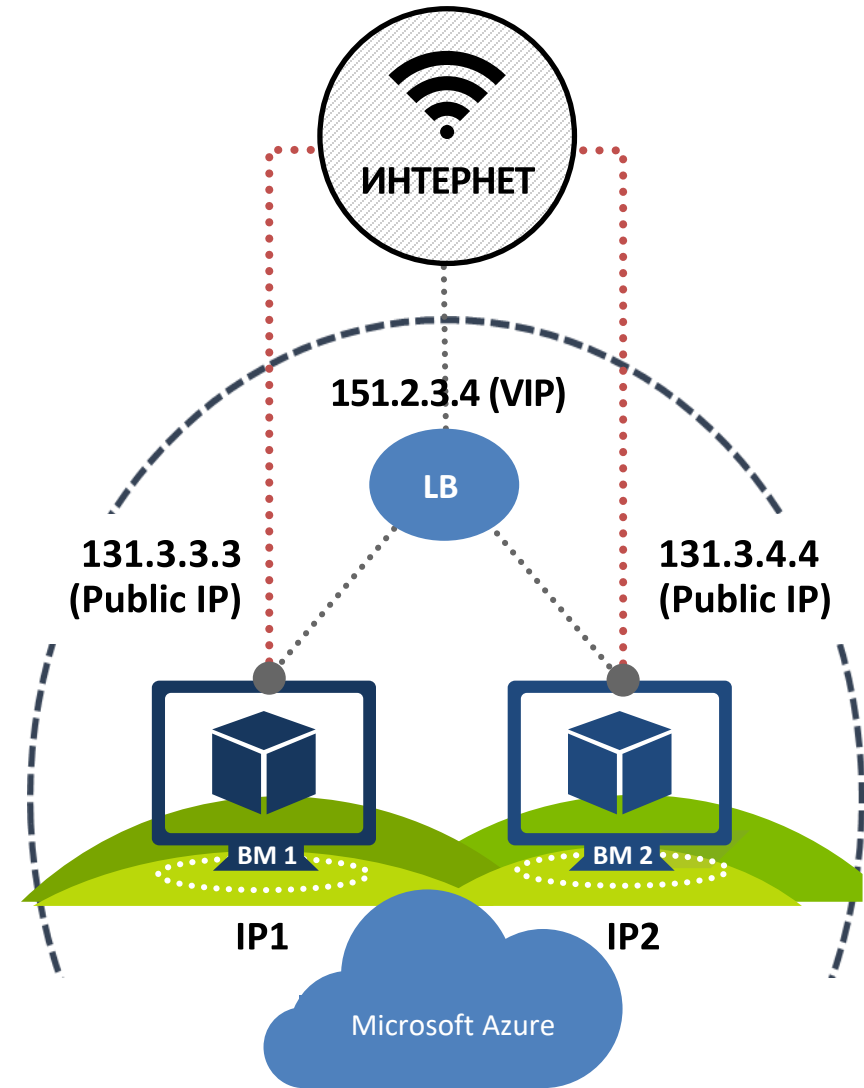
Выделяется динамически (по умолчанию) или статически

## IP для балансировкой нагрузки (VIP)

IP-адрес для балансировкой нагрузки одного и более экземпляров VM

Перенаправление портов

В основном, для высокодоступных сценариев с балансировкой нагрузки или автоматическим масштабированием



# IP-адреса и разрешение имен в ARM

## Частные IP-адреса в Azure

Присваиваются VM, внутренним балансировщикам, шлюзам приложений

## Private IP для VM

IP-адрес из диапазона виртуальной подсети

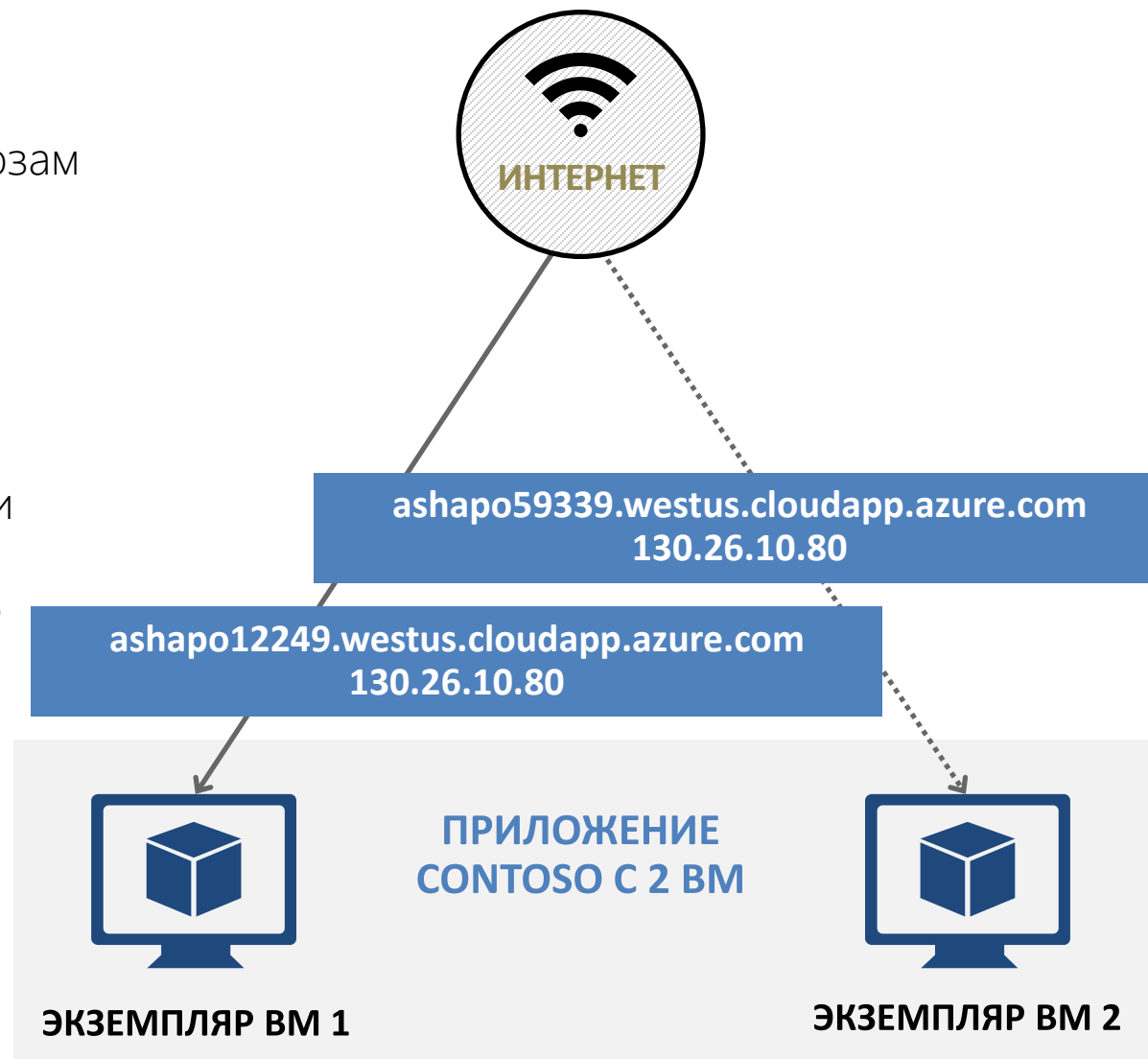
Выделяется динамически (по умолчанию) или статически

## Разрешение имен с помощью Azure DNS

Private IP разрешаются в пределах виртуальной сети

Public IP могут быть присвоены имена *domainnamelabel.location.cloudapp.azure.com*

Имена должны быть уникальны в пределах location



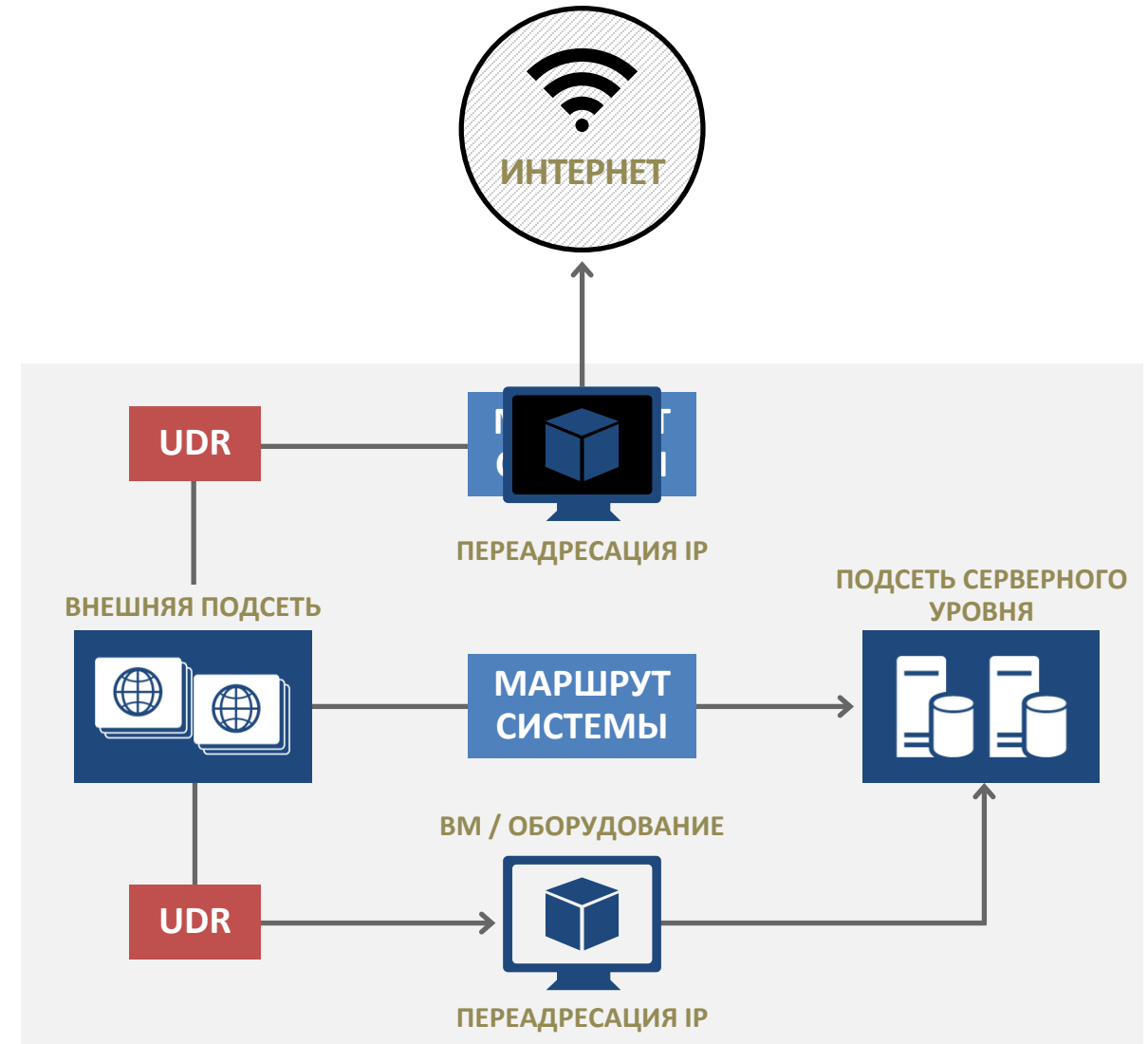
# Пользовательские маршруты (UDR)

Контролируйте сетевой трафик с помощью пользовательских маршрутов

Назначайте подсетям таблицы маршрутов

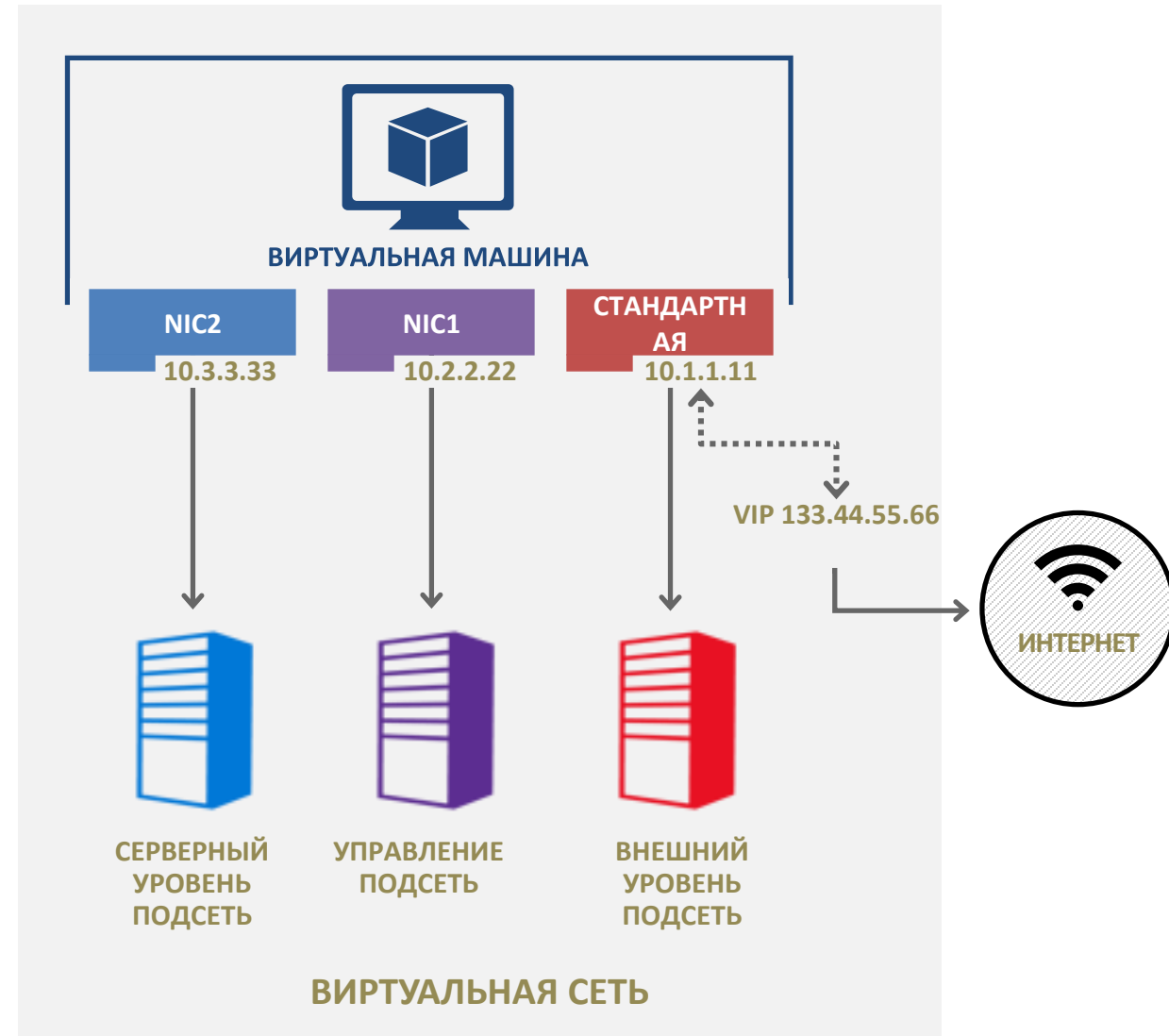
Указывайте следующий сетевой сегмент для любого префикса адреса

Задайте маршрут 0/0 для принудительного туннелирования трафика



# VM с несколькими NIC в Azure

- До 16 NIC на одну VM
- NSG и маршруты на всех NIC
- Разделение внешней подсети, подсети серверного уровня и уровня управления





# Выбор правильной модели подключения

## ШЛЮЗЫ ИНТЕРНЕТ / VPN



- ➔ Подключение через зашифрованное соединение общедоступного Интернета

ПОДКЛЮЧЕНИЕ ЧЕРЕЗ ИНТЕРНЕТ

## ПОСТАВЩИК УСЛУГ



- ➔ Подключение к Azure с помощью услуги ExpressRoute, предоставляемой партнером на своей площадке

EXPRESSROUTE –  
ПРЕДОСТАВЛЯЕТ ПОЛЬЗОВАТЕЛЮ ВОЗМОЖНОСТЬ ВЫБОРА С ДОСТУПОМ КО ВСЕМ ОБЛАЧНЫМ  
СЛУЖБАМ МАЙКРОСОФТ

## ПОСТАВЩИК УСЛУГ



- ➔ Подключение из глобальной сети поставщика сетевых услуг
- ➔ Azure становится одной из площадок пользователя в глобальной сети

# VPN-шлюзы для виртуальной сети

Для доступа к виртуальной сети необходим шлюз ExpressRoute или шлюз VPN

Возможно использование различных SKU

Поддержка совместной работы ExpressRoute и VPN

Повышенная пропускная способность ExpressRoute

SKU ШЛЮЗА ВИРТУАЛЬНОЙ СЕТИ	ПРОПУСКНАЯ СПОСОБНОСТЬ ШЛЮЗА EXPRESSROUTE	ШЛЮЗ VPN – EXPRESSROUTE СОВМЕСТНАЯ РАБОТА	ПРОПУСКНАЯ СПОСОБНОСТЬ ШЛЮЗА VPN	ШЛЮЗ VPN МАКС. ТУННЕЛЕЙ IPSec	ЗАТРАТЫ (В ДОЛЛАРАХ США) В ЧАС
BASIC	500 Мбит/с	НЕТ	100 Мбит/с	10	0,04
STANDARD	1000 Мбит/с	ДА	100 Мбит/с	10	0,19
PERFORMANCE	2000 Мбит/с	ДА	200 Мбит/с	30	0,49

СЛЕДУЕТ ИМЕТЬ В ВИДУ, ЧТО ТРАФИК EXPRESSROUTE ОБЩЕДОСТУПНЫХ СЕРВИСОВ AZURE, O365 И SKYPE ДЛЯ БИЗНЕСА НЕ ПРОХОДИТ ЧЕРЕЗ ШЛЮЗ ВИРТУАЛЬНОЙ СЕТИ

# Группа сетевой безопасности (Network Security Group, NSG)

Сегментация сети для обеспечения безопасности

Набор правил с приоритетами

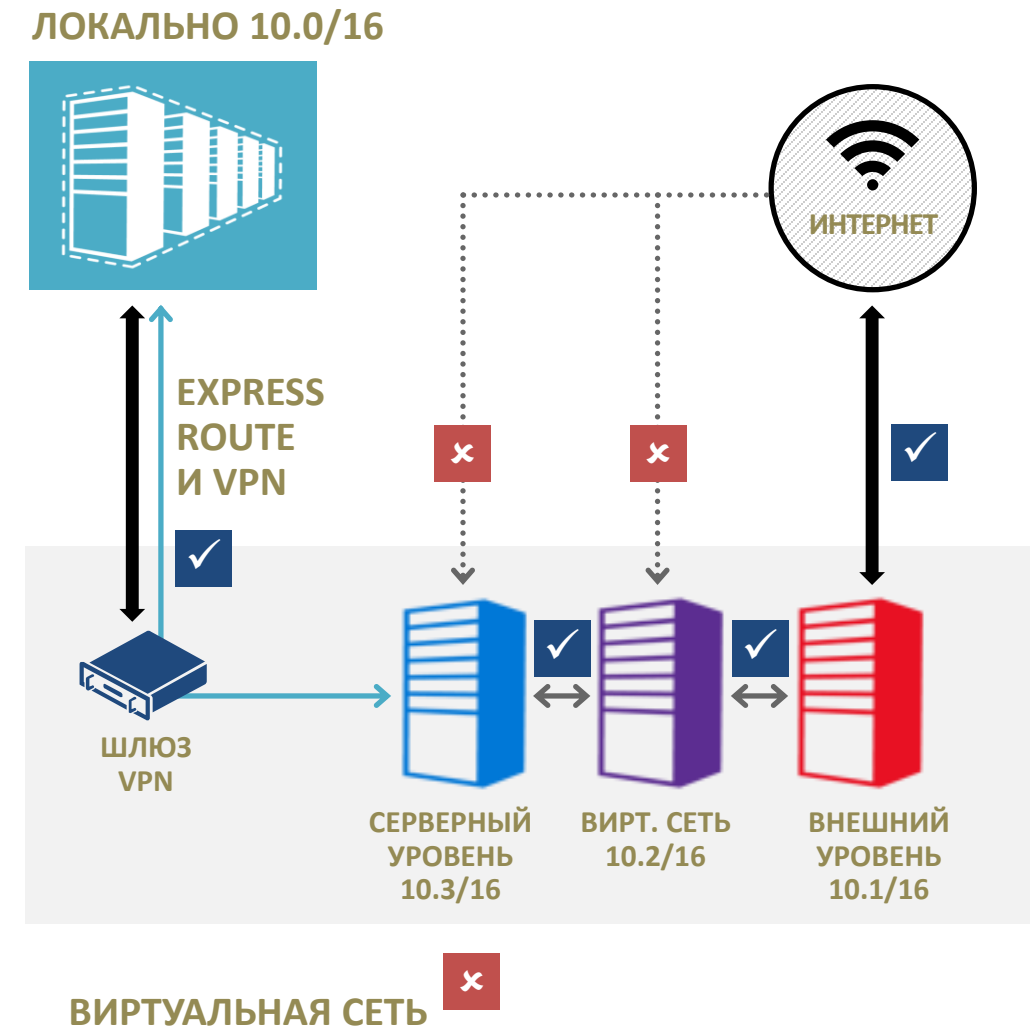
Стандартные правила: 65 000 и более

Применяются к VM и (или) подсети

Применяются ко внутреннему и внешнему трафику

Стандартные тэги: VIRTUAL\_NETWORK, INTERNET, AZURE\_LOADBALANCER

API журналов аудита



# NSG

- Каждая NSG
  - Имеет имя, метку и ассоциирована с регионом
  - Имеет два типа правил, Inbound и Outbound, которые контролируют трафик к VM
    - Правила Inbound применяются к пакетам, приходящим к VM
    - Правила Outbound применяются к пакетам, покидающим VM
    - Входящие и исходящие пакеты должны соответствовать действию 'Allow', иначе они отбрасываются
    - Правила обрабатываются в соответствии с приоритетом, при этом чем меньше номер, тем выше приоритет
    - Если обнаружено соответствие, обработка правил прекращается
  - Может быть ассоциирована с виртуальной сетью, подсетью или VM внутри сети
- VM может быть ассоциирована только с одной NSG, но каждая NSG может содержать до 200 правил

Замечание – endpoint-based ACL и NSG не поддерживаются для одного и того же экземпляра

Замечание – NSG не совместимы с VNETS, ассоциированными с территориальной группой (affinity group)

# NSG – Правила

- Структура правила:
  - **Name:** уникальный идентификатор правила
  - **Type:** Inbound/Outbound
  - **Priority:** целое значение от 100 до 4096
  - **Source IP Address:** IP-адрес отправителя в нотации CIDR
  - **Source Port Range:** целое значение от 0 до 65536
  - **Destination IP Range:** IP-адрес получателя в нотации CIDR
  - **Destination Port Range:** целое значение от 0 до 65536
  - **Protocol:** TCP, UDP или '\*'
  - **Access:** Allow/Deny
- ICMP не может быть указан, но он разрешен в рамках виртуальной сети
- Можно указать диапазон портов, например 100-500
- Правила могут быть изменены в любое время

# NSG – Правила по умолчанию

## Default Inbound Rules

Name	Priority	Source IP	Source Port	Destination IP	Destination Port	Protocol	Access
ALLOW VNET INBOUND	65000	VIRTUAL_NETWORK	*	VIRTUAL_NETWORK	*	*	ALLOW
ALLOW AZURE LOAD BALANCER INBOUND	65001	AZURE_LOADBALANCER	*	*	*	*	ALLOW
DENY ALL INBOUND	65500	*	*	*	*	*	DENY

## Default Outbound Rules

Name	Priority	Source IP	Source Port	Destination IP	Destination Port	Protocol	Access
ALLOW VNET OUTBOUND	65000	VIRTUAL_NETWORK	*	VIRTUAL_NETWORK	*	*	ALLOW
ALLOW	65001	*	*	INTERNET	*	*	ALLOW

 Демонстрация

# Контроль трафика с помощью NSG

#msdevcon

# Контроль доступа к ресурсам ARM



# Контроль доступа в Azure Resource Manager

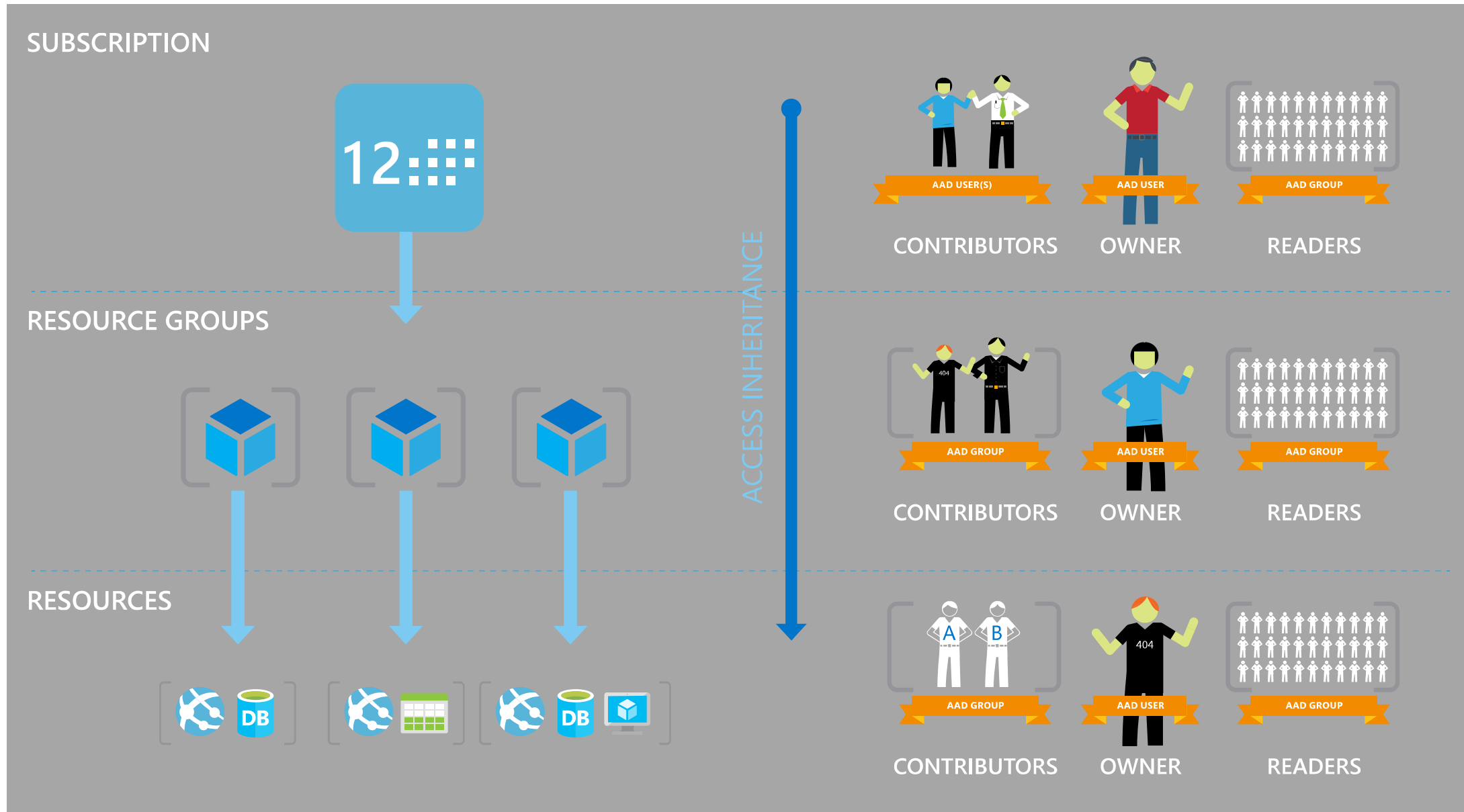
- Механизм ролей (Role Based Access Control, RBAC)
- Журналы аудита
- Блокировки ресурсов

# Role Based Access Control

- Обеспечивает безопасный доступ с гранулярным разграничением полномочий
- Настраивается для пользователей, групп или сервисов
- Встроенные роли упрощают использование

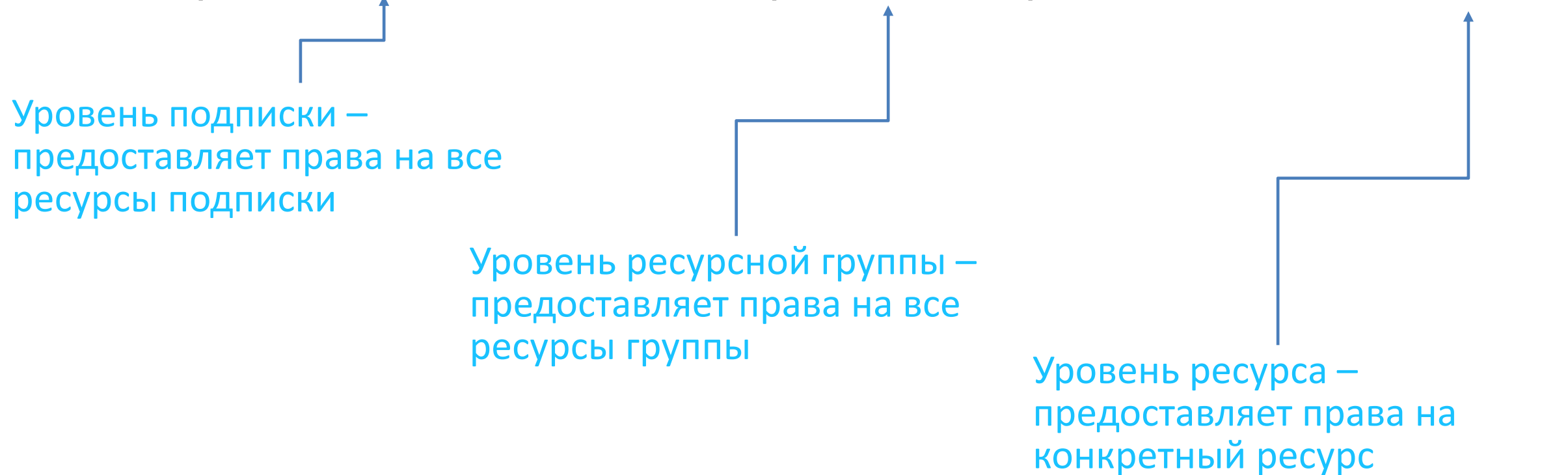
Замечание – RBAC применяется только к ресурсам ARM. ПО внутри VM может иметь собственные механизмы защиты, которые следует учитывать при планировании

# Role Based Access Control



# Уровни гранулярности

/subscriptions/{id}/resourceGroups/{name}/providers/.../sites/{site}



Уровень подписки –  
предоставляет права на все  
ресурсы подписки

Уровень ресурсной группы –  
предоставляет права на все  
ресурсы группы

Уровень ресурса –  
предоставляет права на  
конкретный ресурс

# Журналы аудита

- Регистрируются все операции  
записи/удаления/выполнения
- Централизованное расположение
- Единый формат

# Блокировки ресурсов

- Блокировки ресурсов позволяют предотвратить аварийные ситуации
- Блокировки ресурсов позволяют администратору создавать политики, которые запрещают операции записи или удаления

 Демонстрация

# Разделение полномочий с помощью RBAC

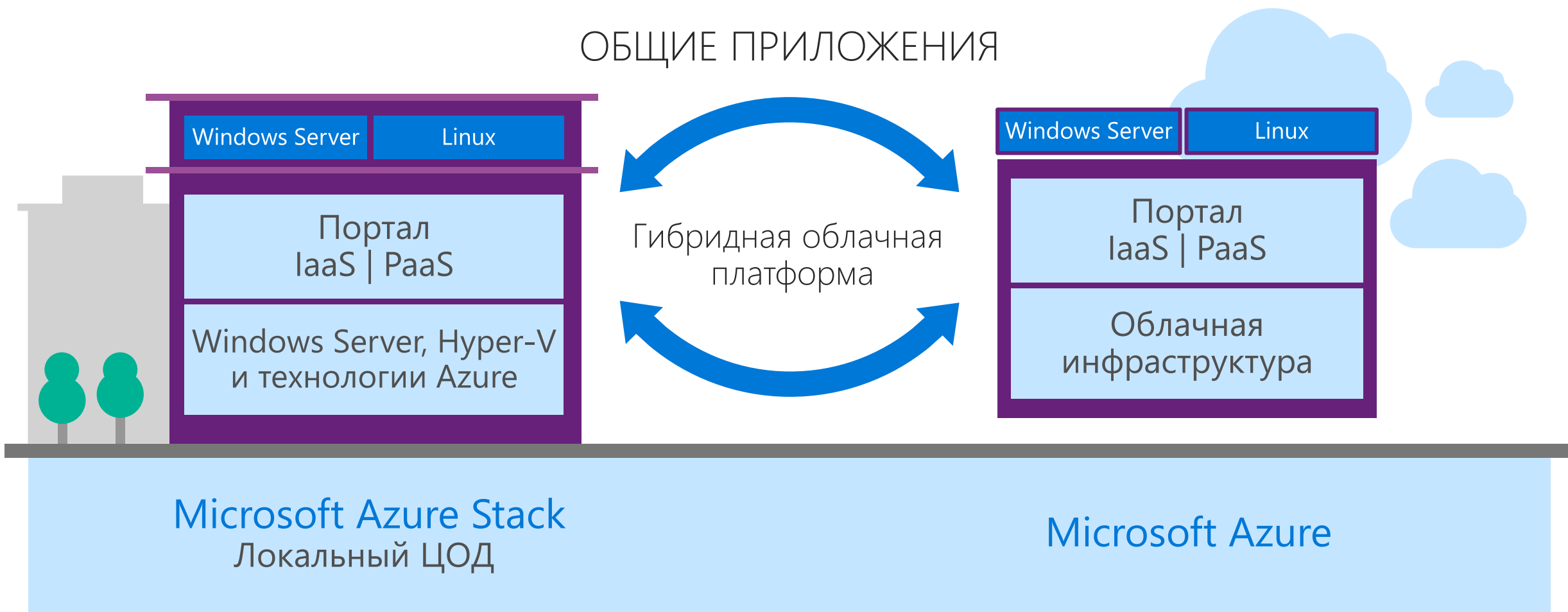
#msdevcon

# ARM и Azure Stack

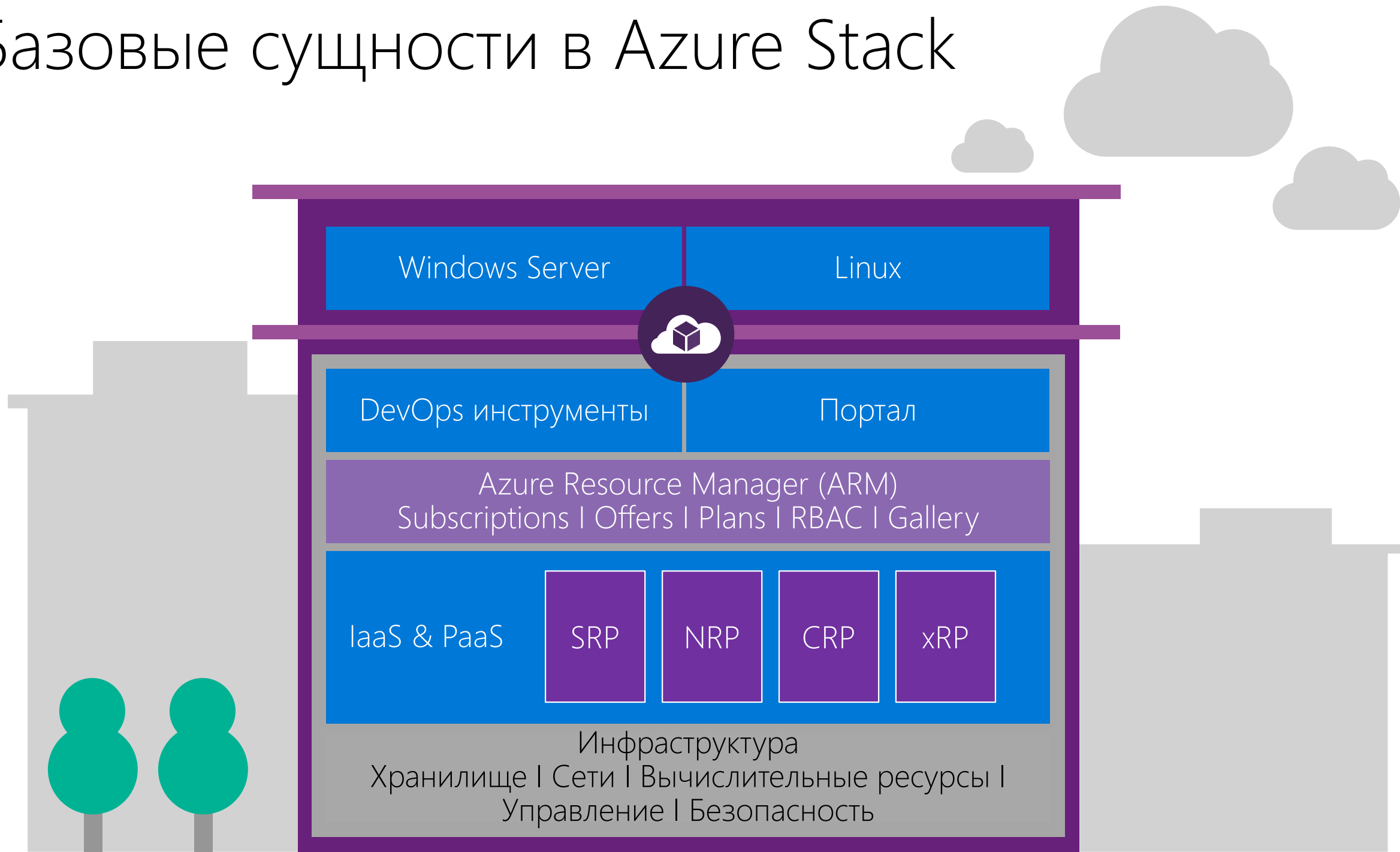
#msdevcon



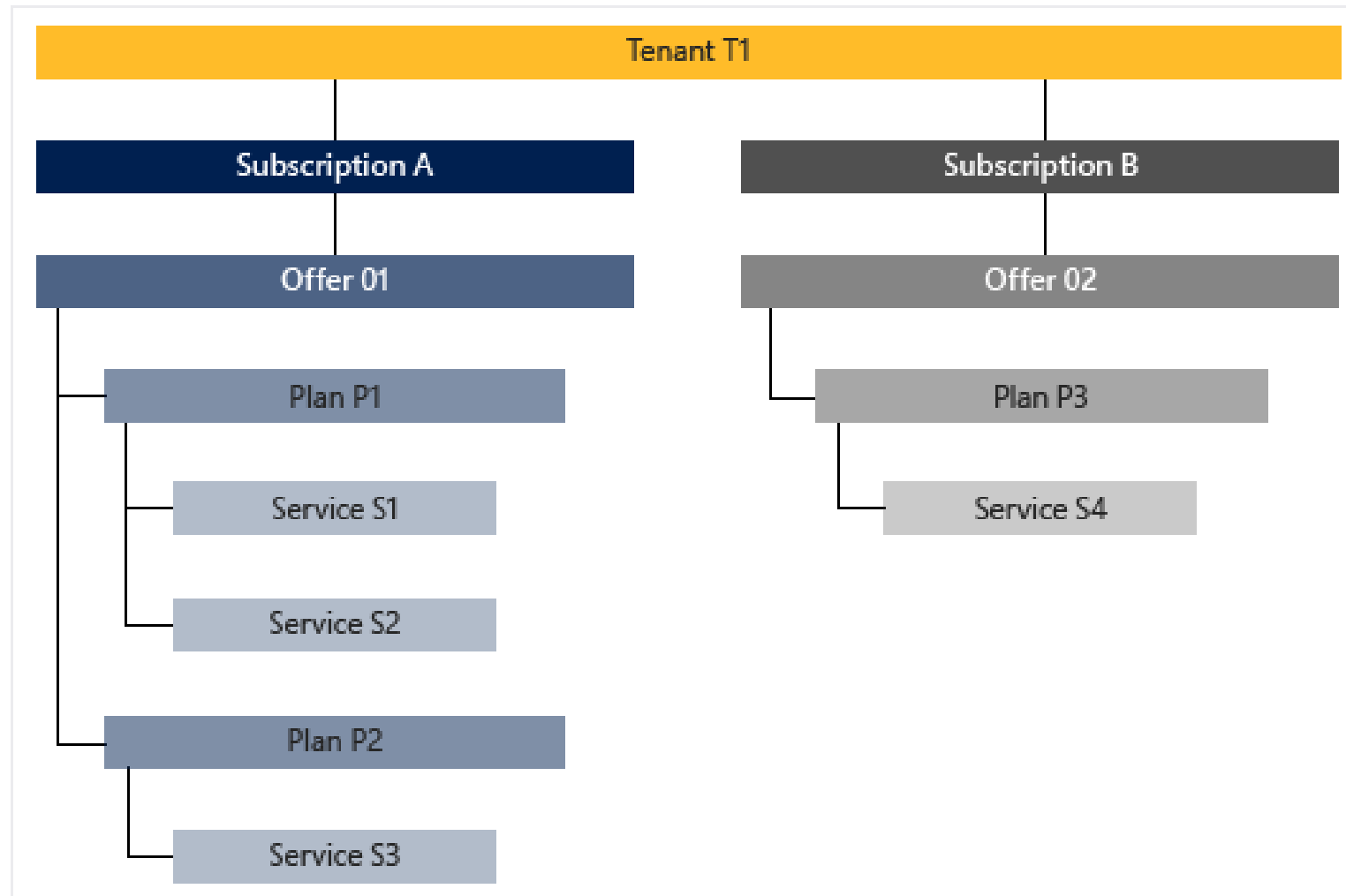
# Azure Stack – сервисы Azure в вашем ЦОД



# Базовые сущности в Azure Stack



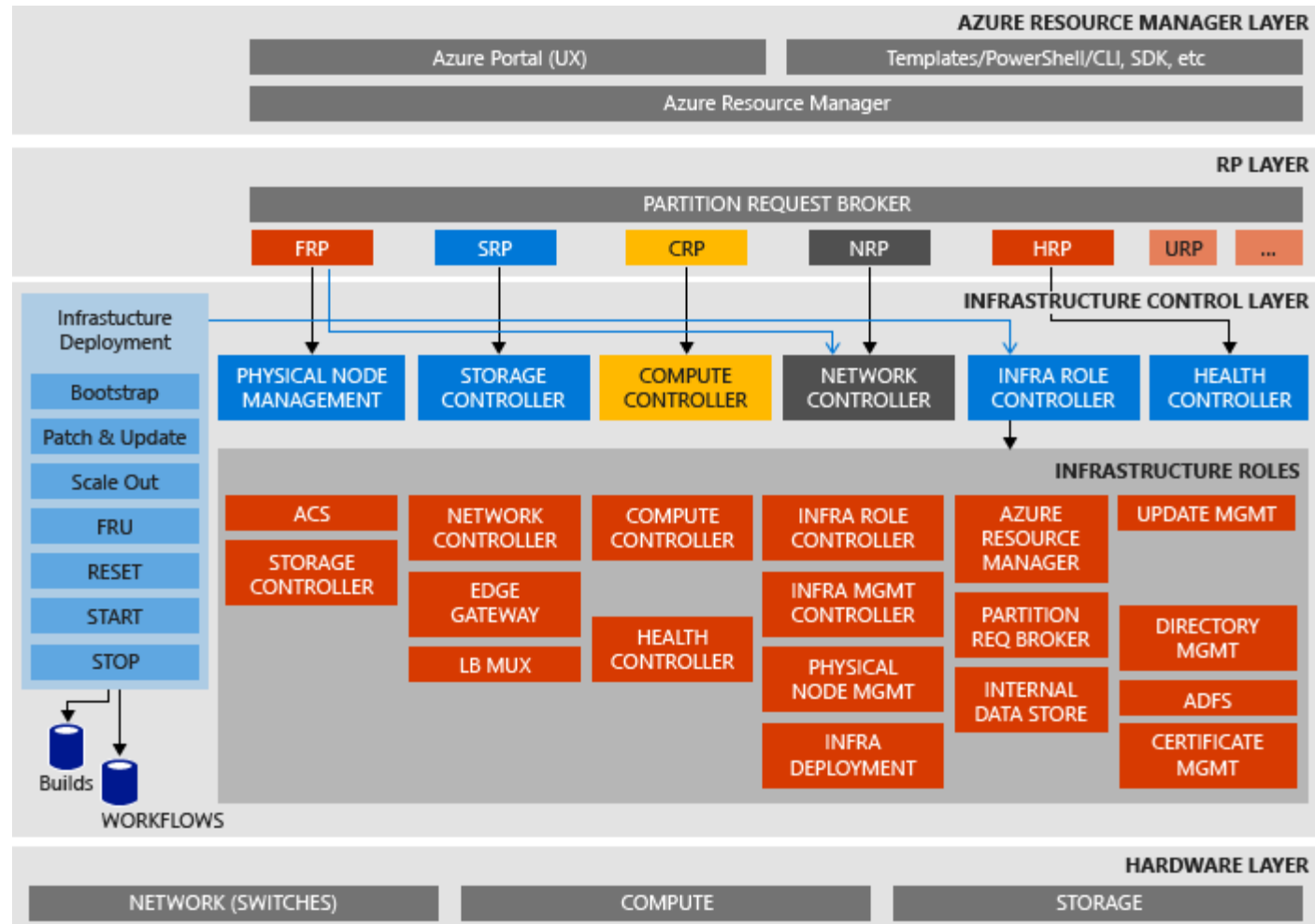
# Базовые сущности в Azure Stack



# Technical Preview 2 - требования

- Azure Active Directory аккаунт с правами Global Admin
  - Можно использовать бесплатный тарифный план FREE
- Аппаратный сервер следующей минимальной конфигурации: 2x CPU суммарно не менее 12 ядер, 96Gb RAM, 5x HDD\SSD: 1x OS 200Gb, 4x 140Gb; RAID "pass-through" или RAID 0 на каждый диск
  - Настраивается загрузка с Windows Server 2016 (vhdx с системой есть в составе установочного пакета)

# Архитектура Azure Stack TP2



## Дополнительные материалы

Документация по ARM

<https://azure.microsoft.com/en-us/documentation/articles/resource-group-overview/>

Deep Dive into Azure Resource Manager Scenarios and Patterns

<https://mva.microsoft.com/en-US/training-courses/deep-dive-into-azure-resource-manager-scenarios-and-patterns-13793>

Инфраструктура в виде услуги на основе Azure Resource Manager

<https://channel9.msdn.com/events/Jump-Starts-Russia/Jump-Start-Infrastructure-As-A-Service-Based-On-Azure-Resource-Manager>

Azure Stack Technical Preview

<https://aka.ms/azurestack/>

#msdevcon

 *Что дальше*

## Используйте ARM

---

Начните использовать ARM  
для реализации  
инфраструктуры в своих  
проектах

## Попробуйте Azure Stack

---

Разверните Azure Stack TP2  
Протестируйте локальные и  
гибридные сценарии на  
базе MAS

## Круглый стол

---

Приходите вечером на  
круглый стол  
Обсудим темы ARM и  
DevOps

#msdevcon



# Использование Azure Resource Manager в управлении жизненным циклом приложений

Александр Шаповал, Microsoft

[ashapo@microsoft.com](mailto:ashapo@microsoft.com)

[@ashapoval](#)

[#msdevcon](#)



# Помогите нам стать лучше!

На вашу почту отправлена индивидуальная ссылка на электронную анкету. 2 ноября в 23:30 незаполненная анкета превратится в тыкву.

Заполните анкету и подходите к стойке регистрации за приятным сюрпризом!

## #msdevcon

Оставляйте отзывы в социальных сетях. Мы все читаем. Спасибо вам! 😊

