

Облачная среда «Асперитас»
Краткое руководство пользователя

Данный документ или его копии не может распространяться (полностью или частично) в любом формате без письменного разрешения ИСП РАН.

1. Описание основных функциональных возможностей облачной среды «Асперитас»

«Асперитас» предназначен для предоставления вычислительных ресурсов по запросу, обеспечивающих отказоустойчивое и высокомасштабируемое функционирование прикладного программного обеспечения различного назначения в единой системе виртуализации за счет:

- сочетания вычислительных средств обработки, хранения и передачи данных в рамках типовой аппаратно-программной платформы;
- возможности гибкого наращивания объёмов различных аппаратных средств путём соединения большого количества типовых узлов и модулей в единую вычислительную систему;
- платформы виртуализации вычислительных средств с централизованным управлением для прикладного программного обеспечения различного назначения;
- возможности динамического перераспределения аппаратных ресурсов по требованию для различных видов прикладного программного обеспечения;

«Асперитас» предназначен для предоставления пользователям виртуальной инфраструктуры (ВИ) с возможностью автоматизированного управления ее жизненным циклом в режиме самообслуживания с возможностью доставлять услуги разных категорий по следующему перечню:

1. «Инфраструктура как услуга» (Infrastructure as a Service, IaaS). Предоставление мощностей в виде одного или нескольких видов ресурсов:
 - вычислительной мощности серверов;
 - емкости систем хранения;
 - пропускной способности каналов.
2. «Платформа как услуга» (Platform as a Service, PaaS). Предоставление вычислительных ресурсов в виде готовых базовых платформ (операционных систем, систем управления базами данных, средств разработки и т.д.) для размещения на них прикладных приложений.

Для функционирования программных компонентов необходимы следующие технические средства:

узел развертывания (сервер или рабочая станция):

- 1) процессор с числом ядер не менее 8 с архитектурой x86 или x64;
- 2) не менее 64 ГБ оперативной памяти;
- 3) 2 сетевых интерфейса с пропускной способностью не менее 10 Гбит/с;
- 4) не менее 150 ГБ свободного места на жестком или SSD-диске.

контроллер (3 сервера для масштабов 1-20 узлов виртуализации):

- 1) процессор с числом ядер не менее 32 с архитектурой x86 или x64;
- 2) не менее 192 ГБ оперативной памяти;
- 3) 2 сетевых интерфейса с пропускной способностью не менее 10 Гбит/с;
- 4) не менее 512 ГБ свободного места на жестком или SSD-диске;
- 5) выделенный интерфейс IPMI не ниже 5 версии.

узел виртуализации (сервер или рабочая станция):

- 1) поддержка аппаратной виртуализации;
- 2) процессор с числом ядер не менее 16 с архитектурой x86 или x64
- 3) не менее 96 ГБ оперативной памяти;
- 4) 2 сетевых интерфейса с пропускной способностью не менее 10 Гбит/с;
- 5) не менее 1 ТБ свободного места на жестком или SSD-диске;
- 6) выделенный интерфейс IPMI, не ниже 5 версии.

узел хранения данных (сервер или рабочая станция):

- 1) процессор с числом ядер не менее 24 с архитектурой x86 или x64;
- 2) не менее 128 ГБ оперативной памяти;
- 3) 2 сетевых интерфейса с пропускной способностью не менее 10 Гбит/с;
- 4) не менее 96 ТБ свободного места на жестком или SSD-диске (объектовое хранилище), не менее 250 ГБ свободного места на жестком или SSD-диске (операционная система), не менее 256 ГБ свободного места на NVMe/M.2/SATA SSD-диске (журналы).

коммутатор:

- 5) 24 сетевых интерфейса с пропускной способностью не менее 10 Гбит/с;

6) поддержка технологии 802.1q.

Облачная среда «Асперитас» разрабатывается на языках программирования:

- 1) Python, для создания объектного кода используется открытое программное обеспечение языка Python (лицензия Python Software Foundation License версия 2);
- 2) C для создания объектного кода используется открытое программное обеспечение, компилятор GCC (лицензия GNU General Public License версия 3).

Облачная среда Asperitas разработана на базе программного обеспечения с открытым исходным кодом Openstack (лицензия Apache версии 2.0) и использует в своей работе программное обеспечение с открытым исходным кодом Serp (лицензия LGPL версия 3.0).

2. Лицензирование и установка

По вопросам определения стоимости, приобретения и использования, обращайтесь по адресу asperitas@ispras.ru.

2.1. Установка облачной среды

2.2.1. Область применения

К основным задачам, которые способен решать облачный дистрибутив «Асперитас», относятся:

- развертывание частных облачных сред на физических серверах в условиях отсутствия подключения к интернету;
- развертывание надежной системы хранения.

2.2.2 Схема развертывания облачной среды «Асперитас»

Развертывание состоит из следующих шагов (более подробно они будут описаны в пункте «Инструкция по развертыванию облачной среды»):

1. Подготовка аппаратного оборудования, настройка сетевого взаимодействия для корректного развёртывания облачной среды «Асперитас».
2. Определение ролей узлов, выделение узла развёртывания.
3. Установка системы с диска облачного дистрибутива «Асперитас»
4. Запуск сценария установки сервисов, необходимых для узла

развёртывания.

5. Заполнение файла *instackenv.yaml*, указывающего на узлы, предназначенные для облачной среды «Асперитас».
6. Если о какой-либо из машин неизвестны параметры количества ЦПУ, размера памяти дисковой и оперативной, то необходимо провести предварительный сбор данных об узлах с узла развёртывания.
7. Конфигурация развёртывания системы хранения данных; на данном этапе необходимо настроить параметры настройки сетей и работы с блочными устройствами;
8. Запуск процесса развёртывания надёжной файловой системы;
9. Запуск процесса развёртывания облачной среды.

Для полного функционирования облачной среды «Асперитас» необходим следующий набор физических узлов:

- узел развёртывания. Представляет собой узел, необходимый на этапах развёртывания остальных частей облачной среды «Асперитас», а также при обновлении и изменении остальных функциональных частей;
- узлы хранения. Содержат множество дисков и большое количество дискового пространства, а также представляют собой систему надёжного хранения пользовательских данных в облачной среде «Асперитас». Количество должно составлять не менее 4-х;
- узлы вычислительные. Содержат все запущенные ВМ. Количество их оперативной памяти (ОП), дискового пространства и ЦПУ является вычислительной мощностью облачной среды «Асперитас»;
- узлы управления. Являются центром управления вычислительными узлами и центром общения с узлами хранения. При полном выводе из строя узлов управления запущенные ВМ продолжают свое функционирование.

Узлы управления предоставляют пользователю контроль над облачными ресурсами: их созданием, изменением и удалением. Узлы хранения позволяют надёжно хранить информацию на дисках, созданных пользователем. Вычислительные узлы являются основным нагрузочным элементом, так как содержат динамически распределённые виртуальные машины пользователей.

2.2.3 Основные возможности инструментов развёртывания Асперитас

Функциональными элементами облачной среды «Асперитас» являются:

- распределённая сеть хранения данных;

- системы хранения облачных образов, настройки виртуальных сетей, запуска VM и др. на базе OpenStack;
- инструмент мониторинга.

Облачный дистрибутив «Асперитас» позволяет автоматизированно развертывать указанные выше функциональные элементы, а также централизованно с узла развертывания проводить обновление элементов и изменение элементов при необходимости.

Инструмент мониторинга предоставляет централизованное отслеживание сбоев и ошибок на узлах, а также централизованное отслеживание статуса VM, запущенных в облачной среде «Асперитас».

2.2.4 Первоначальное развертывание

При первоначальном развёртывании строго рекомендуется пройти каждый пункт! Asperitas Console не предполагает быстрого старта или прототипного развёртывания!

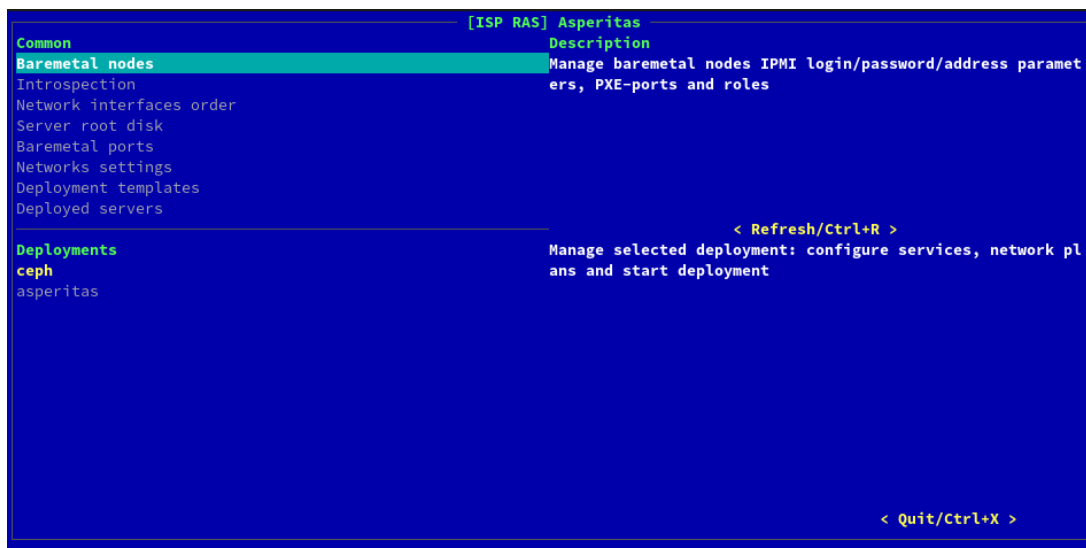
Для запуска консоли введите команду в консоли узла развёртывания

```
$ asperitas
```

Для запуска консоли в режиме **debug** используйте команду

```
$ asperitas --debug
```

После выполнения команды появляется **Главное окно** asperitas консоли



2.2.5 Физические узлы

Перейдите во вкладку *Baremetal nodes*. Нажмите кнопку *Add*, чтобы добавить новый физический узел для добавления в развёртываемое облако.

Baremetal node Details

Name: [Redacted]

BMC type: [IPMI]

BMC address: [Redacted]

BMC login: [Redacted]

BMC password: [Redacted]

Profile: [Controller]

UEFI boot mode: [] set if UEFI boot mode is used

PXE port mac_address: [Redacted]

Disk capacity (Gb): [Redacted]

Cpu number: [Redacted]

Architecture: x86_64

Memory capacity (Mb): [Redacted]

< Save/Ctrl+U > < Cancel/Ctrl+C >

Красным цветом выделены поля обязательные для заполнения.

- *Name* - имя физического узла. Используется лишь для удобства, не влияет на hostname узла;
- *BMC type* - тип BMC, поддерживаемые значения: *IPMI*, *iDRAC*;
- *BMC address* - IP-адрес BMC, пример: 2.3.4.5 ;
- *BMC login* - логин для управления физическим узлом через BMC;
- *BMC password* - пароль для управления физическим узлом через BMC;
- *Profile* - роль физического узла в развёртываемом облаке;
- *UEFI boot mode* - установите галочку, если узел загружается в режиме UEFI;
- *PXE port mac_address* - мак адрес физического порта узла, через который происходит загрузке по PXE.

Поля необязательные для заполнения (используются, если интроспекция узла нежелательна):

- *Disk capacity* - размер физического диска для установки операционной системы;
- *Cpu number* - число процессорных ядер узла;
- *Architecture* - архитектура процессора (поддерживается только x86_64);
- *Memory capacity* - размер оперативной памяти узла.

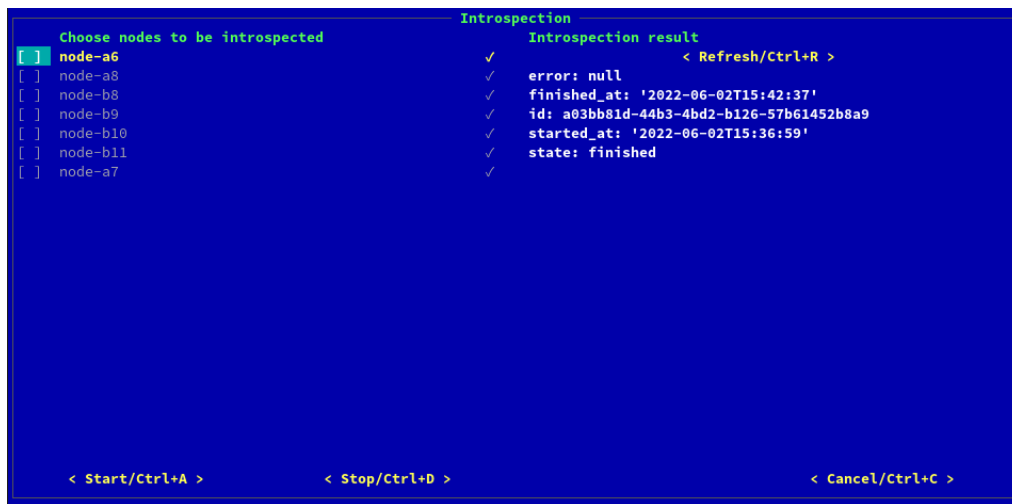
Нажмите кнопку *Save*. Физический узел добавится в список физических узлов. При добавлении физического узла узел не включается и вводится в развёртываемое облако.

Дождитесь изменения статуса узла на *manageable/available*. Если узел остаётся в статусе *enroll*, перейдите в раздел «Исправление проблем».

Статус *manageable* позволяет включать и выключать узел из системы на узле развёртывания. Статус *available* позволяет ввести машину в развёртываемое облако. Система автоматически переводит узел в состояние *available*, если известны параметры *Disk capacity*, *Cpu number*, *Architecture* и *Memory capacity*.

2.2.6 Интроспекция

Перейдите во вкладку *Introspection* Главного меню.



На экране отобразится список физических машин, добавленных в предыдущем пункте.

Отметьте весь список машин и нажмите кнопку *Start*. Начнётся интроспекция физических узлов. Статус отображается в правой половине экрана.

После успешного окончания интроспекции напротив поле *state* будет равно значению *finished*, а напротив имени узла появится галочка.

Статус отображается только для одного узла. Чтобы посмотреть статус для другого узла, нажмите на имя узла.

После успешной интроспекции можно посмотреть собранные данные дважды нажав мышкой на имя узла. Если мышки нет, то при помощи кнопок наведите на имя узла и нажмите *Enter*

```

Introspection
bmc_address: 10.100.2.20
bmc_v6address: ::/0
boot:
  current_boot_mode: bios
  pxe_interface: ac:1f:6b:1a:6b:9a
cpu:
  architecture: x86_64
  count: 32
  flags:
    - fpu
    - vme
    - de
    - pse
    - tsc
    - msr
    - pae
    - mce
    - cx8
    - apic
    - sep
    - mtrr
    - pge
    - mca
    < Cancel/Ctrl+C >

```

2.2.7 Порядок физических интерфейсов

! Опция доступна только после успешной интроспекции

Перейдите во вкладку *Network interfaces order*.

```

Introspection
Node node-a8
  nic1 [0x8086:0x15ab eno1 ac:1f:6b:1b:ce:ea ]
  nic2 [0x8086:0x15ab eno2 ac:1f:6b:1b:ce:eb ]
Node node-b8
  nic1 [0x8086:0x15ab eno1 00:25:90:bb:0e:d8 ]
  nic2 [0x8086:0x15ab eno2 00:25:90:bb:0e:d9 ]
Node node-b9
  nic1 [0x8086:0x15ab eno1 00:25:90:bb:0e:be ]
  nic2 [0x8086:0x15ab eno2 00:25:90:bb:0e:bf ]
Node node-b10
  nic1 [0x8086:0x15ab eno1 00:25:90:bb:0d:5c ]
  nic2 [0x8086:0x15ab eno2 00:25:90:bb:0d:5d ]
Node node-b11
  nic1 [0x8086:0x15ab eno1 00:25:90:bb:0d:92 ]
  nic2 [0x8086:0x15ab eno2 00:25:90:bb:0d:93 ]
Node node-a7
  nic1 [0x8086:0x15ab eno1 ac:1f:6b:1a:6b:54 ]
  nic2 [0x8086:0x15ab eno2 ac:1f:6b:1a:6b:55 ]
  < Save/Ctrl+U >
  < Cancel/Ctrl+C >

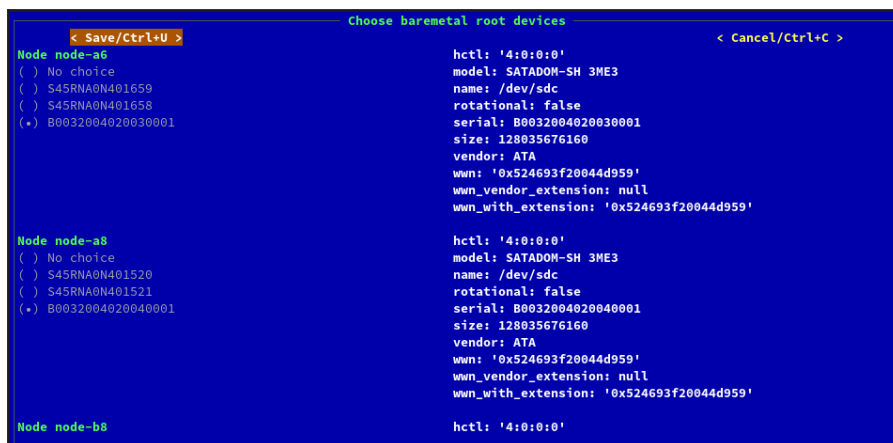
```

На экране отобразится список узлов и их интерфейсов. Порядок интерфейсов важен, так как далее сетевые планы строятся только по номерам интерфейсов, а не их именам. Поэтому определите порядок так, чтобы интерфейсы всех узлов с одним порядковым номером были настроены одинаково на физических коммутаторах в вашей серверной.

2.2.8 Диск операционной системы

! Опция доступна только после успешной интроспекции

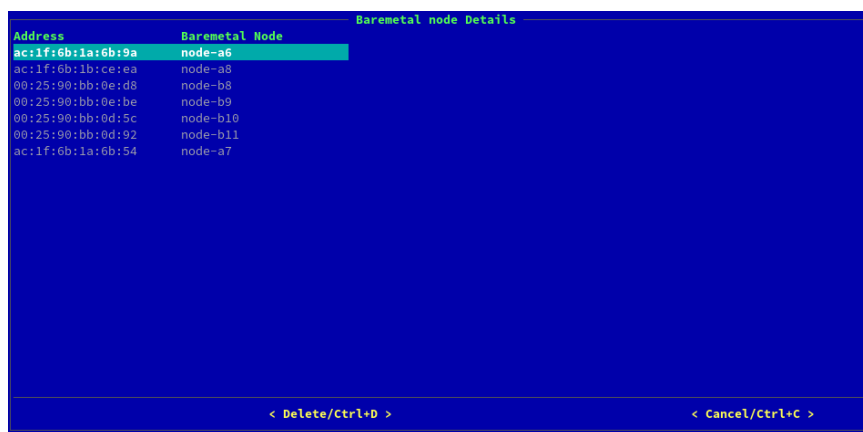
Перейдите во вкладку *Server root disk*.



Если у какого-либо физического узла несколько дисков, то необходимо указать правильный диск для установки операционной системы. Иначе система выберет физический диск случайным образом.

2.2.9 Физические порты

Перейдите во вкладку *Baremetal ports*.



Здесь отображаются все физические порты известные системе для загрузки по PXE. Первоначально создаётся порт указанный при создании физической машины. Затем после интроспекции добавляется порт, по которому произошла загрузка по PXE, **если этот порт не совпал с указанным вами при создании**.

Если это произошло, то необходимо оставить один порт в системе и удалить ненужный! Иначе при развёртывании порт для загрузки будет выбран случайным образом из этих двух.

2.2.10 Сети

Перейдите во вкладку *Networks*.

List OpenStack physical networks						
Name	VLAN	CIDR	Gateway	MTU	Exists	Routes
Ctlplane		192.168.24.0/24	192.168.24.1	1500	[✓]	[]
External	112	10.100.112.0/22	10.100.115.254	1500	[✓]	[{"destination": "10.10.12.0/24", "next
InternalApi	12	172.16.12.0/24		9000	[✓]	[]
Storage	30	10.100.100.0/22		9000	[✓]	[]
InternalCeph	31	172.16.31.0/24		9000	[]	[]
IronicProvisioning	11	172.16.11.0/24		9000	[✓]	[]

< Save/Ctrl+U > < Add/Ctrl+A > < Delete/Ctrl+D > < Cancel/Ctrl+C >

В системе Asperitas используется принцип деления сетей на роли. Набор заранее заданных сетей, который обязан присутствовать в системе:

- *Ctlplane* - сеть администрирования физических узлов в развёртываемом облаке. Это единственная сеть, по которой разрешён SSH доступ к физическим узлам облака.
- *External* - сеть для доступа к публичному API развёртываемого облака. Сеть должна быть доступна для всех пользователей облака.
- *InternalApi* - сеть для доступа к приватному API развёртываемого облака. Используется сервисами облака для общения друг с другом. Не должна быть доступна даже с узла развёртывания.

- *Storage* - сеть для доступа сервисов облака к хранилищу: Ceph или Nfs.

Набор заранее заданных опциональных сетей, их можно удалить из системы за ненадобностью:

- *InternalCeph* - внутренняя сеть для узлов хранения Ceph.
- *IronicProvisioning* - сеть для захвата физических узлов сервисом Ironic развёртываемого облака. Если сервис не используется, то сеть не нужна.

Можно изменить параметры сети, нажав на сеть в списке сетей двумя кликами мыши или кнопкой *Enter*.

```

Modify OpenStack physical network
Name: IronicProvisioning
VLAN ID: 11
CIDR: 172.16.11.0/24
Allocation pools: [{"start": "172.16.11.30", "end": "172.16.11.99"}]
Gateway IP:
Routes: []
MTU: 9000

< Save/Ctrl+U > < Cancel/Ctrl+C >

```

Если необходима дополнительная сеть, то можно добавить новую, нажав кнопку *Add*.

Если в поле *Exists* стоит галочка, то сеть уже создана в системе и её нельзя удалить из списка сетей.

Сеть создаётся при развёртывании в дальнейшем и удаляется, если удаляется развёртывание.

Name	Images Path
ceph	/home/stack/images-ceph
asperitas	/home/stack/images-asperitas

< Add/Ctrl+A > < Delete/Ctrl+D > < Cancel/Ctrl+C >

2.2.11 Тестовый сценарий работы со средой развёртывания

Проверка работоспособности частично проводится во время развёртывания облачной среды «Асперитас» при инициализации сервисов и некоторых облачных ресурсов. Во время развёртывания проходят инициализация баз данных (БД) облачных сервисов, что не позволяет завершить процесс развёртывания при возможных ошибках БД.

Проверяется доступность узлов другими узлами в каждой сети, настроенной на физических узлах. А затем настраивается сервис для высокой доступности сервисов, который гарантирует корректное функционирование сервисов балансировки, БД

Galera, БД OVN, сервиса обмена сообщениями между системами облачной среды RabbitMQ и доступность виртуальных IP адресов.

Создаются проекты, домены, пользователи сервисов и их роли, что невозможно без корректного функционирования сервиса аутентификации.

По завершении развертывания необходимо выполнить с узла развертывания скрипты проверки облачной платформы. Выполнить команды из консоли:

```
$ source overcloudrc
$ cd cloud-prepare-scripts/playbooks
```

Файл `roles/check-nova-network/vars/main.yaml` содержит настраиваемые параметры. Обратите внимание на переменную `network_1` и ее настройки. Для функционирования этой виртуальной сети в облаке необходимо, чтобы на коммутаторах, к которым присоединяются узлы управления и вычислительные узлы, был заведен VLAN 450 в режиме `tagged` для портов узлов управления и вычислительных узлов. А также в сети 10.100.148.0/22 должен был поднят DHCP-сервер с адресом 10.100.151.254, маршрутизирующий пакеты тегированные числом 450 и маршрутизирующий пакеты в сеть Интернет.

Запустить скрипт создания тестовых ресурсов:

```
$ ansible-playbook check-nova-network.yaml
```

При успешном выполнении данного скрипта на экране отобразится сообщение «Execute: ping <test ip>» и IP-адрес тестовой машины вида 10.100.150.*. Чтобы протестировать доступность тестовой машины, добавьте маршрутизацию из сети доступа к узлу развертывания в сеть 10.100.148.0/22. Затем выполните с узла развертывания:

```
$ ping <test ip>
$ ssh cirros@<test ip>
```

Последняя команда подключит вас к консоли тестовой машины, с которой можно выполнить некоторые действия. Например, проверить доступность сети или любого другого адреса:

```
$ ping 8.8.8.8
```

В случае если недоступен IP адрес тестовой машины, то убедиться, что адрес 10.100.151.254. Так как самая частая причина – это неправильная настройка маршрутизации.

Если проблема не в этом, то зайти в графический интерфейс облака по адресу 10.100.144.100. Логин/пароль – admin/пароль, указанный при развертывании системы. Зайти во вкладку «Project – Compute – Instances». Нажать на `BM testing_config` и затем

выбрать вкладку VNC. Так как это тестовая машина, то она запущена из системы Cirros с заранее заданными пользователем/паролем, указанными на экране.

Проверить, что VM получила имя хоста, равное `testing_config`. Получила внутренний адрес из сети 192.168.33.0/24 и маршрут по умолчанию через 192.168.33.1. При отсутствии любого из пунктов необходимо посмотреть логи neutron на узлах управления и вычислительных и устранить указанные там ошибки. А также проверить, что все завершенные контейнеры завершены со статусом 0.

В случае если вышеуказанное работает, то ошибку искать в связности и настройках сети, соответствующей VLAN 450.

Если ошибка произошла при создании образа Cirros, то посмотреть логи Glance на узлах управления и исправить указанные ошибки.

Если ошибка произошла при создании VM, то посмотреть логи Nova на узлах управления и вычислительных узлах и исправить указанные ошибки.

В предыдущих двух случаях также возможны ошибки со стороны системы хранения. Если в логах указаны ошибки обращения к узлам хранения, то посмотреть логи на узлах хранения и исправить указанные ошибки.

2.2. Инструкция по развертыванию облачной среды

Ниже описаны требования к физическим устройствам и настройкам сети, необходимым для развертывания облачной среды, а также содержится инструкция по развертыванию облака контейнеров и облака VM с использованием распределенной ФС.

2.3.1. Требования к настройке сетей на физических машинах

На всех узлах необходимо наличие сетей, указанных на рис. 1. Наличие каждой из сетей имеет свои свойства и требования. На рис. 1 не указаны сети IPMI. Управление через IPMI требуется от узлов хранения, управления и вычислительных узлов. От узла развертывания требуется доступ к этой сети для управления указанными узлами.

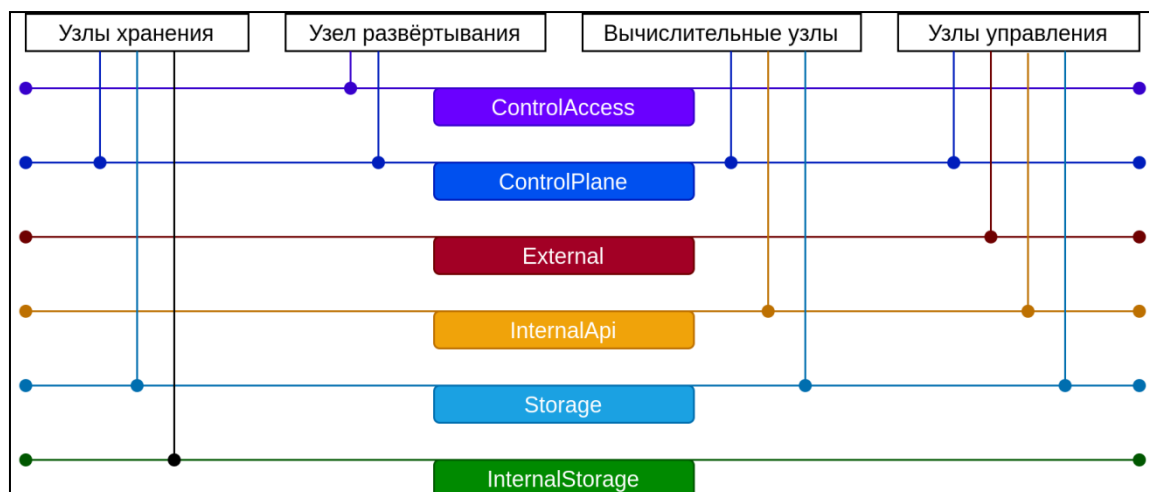


Рис. 1. Схема используемых сетей

Узел развертывания должен иметь два интерфейса, один интерфейс настраивается для доступа к узлу из сети предприятия. Ограничений для этой сети нет. И второй интерфейс должен быть без каких-либо настроек сети. Порт соответствующего интерфейса на коммутаторе должен находиться в режиме untagged для некоторого ControlPlaneVLAN, например, 10.

Узлы управления и вычислительные узлы должны иметь по два интерфейса, каждый из которых также имеет соответствующий порт на коммутаторе в режиме hybrid с ControlPlaneVLAN в качестве access. То есть весь нетегированный трафик, проходящий через эти интерфейсы на коммутаторе должен помечаться ControlPlaneVLAN.

Узел хранения должен иметь один интерфейс в режиме hybrid с ControlPlaneVLAN в качестве access и еще три интерфейса в режиме untagged для некоторого InternalStorageVLAN.

На коммутаторах, связывающих узлы управления, хранения и вычислительные узлы завести ExternalVLAN, InternalAPIVLAN, StorageVLAN и добавить их соответствующим портам исходя из схемы сетей на рисунке 1.

Сеть сопровождения облака (ControlPlane). Эта сеть настраивается на узле развертывания на этапе настройки и поднятия сервисов, необходимых для развертывания остальной части облачной среды «Асперитас». После этого на узле развертывания поднимается DHCP-сервер для раздачи адресов остальным узлам облачной среды из сети 192.168.24.0/24.

IP-адрес узла развертывания в этой сети – 192.168.24.1, 192.168.24.2, 192.168.24.3.

Сеть используется для общения между узлом развертывания и облачной

средой в процессе развертывания. В дальнейшем используется как служебная сеть облачной среды «Асперитас».

Внешняя сеть доступа к облачной среде «Асперитас» (External). В сети должен существовать маршрутизатор с адресом 10.100.147.254. По этой сети пользователи облачной среды получают публичный доступ к облачным сервисам. Используемый для доступа основной адрес – 10.100.144.100

Сеть с зоной видимости внутри облачной среды для внутренних коммуникаций (InternalAPI) 172.16.2.0/24.

Сеть для доступа облачных сервисов к системе хранения (Storage) 172.16.1.0/23, адреса из этой сети – точки входа в распределенной системе хранения для клиентов.

Сеть настройки системы хранения и синхронизации – (InternalStorage) 172.16.3.0/24.

Для функционирования надежной ФС необходимо наличие двух сетей – внутренней и публичной. Рекомендуемая пропускная способность для обеих сетей 10 Гбит/с или выше. Также рекомендуется поддержка больших размеров кадра (jumboframes) на коммутационном оборудовании.

2.3.2. Подготовка узла развертывания

Узел развёртывания обязан иметь два интерфейса. Интерфейс для доступа к узлу настраивается исходя из требований секретности компании.

Подключитесь к виртуальной машине развёртывания:

```
$ ssh stack@IPADDR
```

Второй интерфейс, выделенный для сети ControlPlane, пусть называется eno1. Необходимо открыть для редактирования файл undercloud.conf и внести запись, например:

```
DEFAULT]
local_interface=enol
```

Выполните следующую команду для настройки узла развёртывания (команда может выполняться 20-50 минут в зависимости от технических характеристик физического узла):

```
$ openstack undercloud install
```

2.3.3 Развертывание распределенной файловой системы

На узлах распределенной файловой системы необходимо выполнить установку пакета openssh-server.

На каждом узле необходимо создать пользователя с именем `cephuser`, обладающего `sudo`-правами.

При помощи утилиты `ssh-copy-id` скопировать публичные ключи на узлы распределенной файловой системы:

```
$ ssh-copy-id -i id_rsa.pub cephuser@<host>
```

Запустить сервис `ntp` и синхронизировать время

Основные файлы конфигурации:

1. `ceph_hosts`
2. `ansible-master-distro/group_vars/all.yml`

Файл `ceph_hosts` содержит информацию об IP-адресах узлов, на которых будет производиться развертывание.

Список устройств, используемых для распределенной файловой системы, прописывается для каждого узла отдельно в файле `hosts` в секции `osds`. Список устройств необходимо заменить на соответствующий

```
[osds]
10.10.12.76    devices="['/dev/sdc', '/dev/sdb', '/dev/sde']"
10.10.12.77    devices="['/dev/sdc', '/dev/sdb', '/dev/sde']"
10.10.12.78    devices="['/dev/sdc', '/dev/sdb']"
```

В файле конфигурации распределенной файловой системы `all.yml` описываются настройки публичной и внутренней сети распределенной файловой системы.

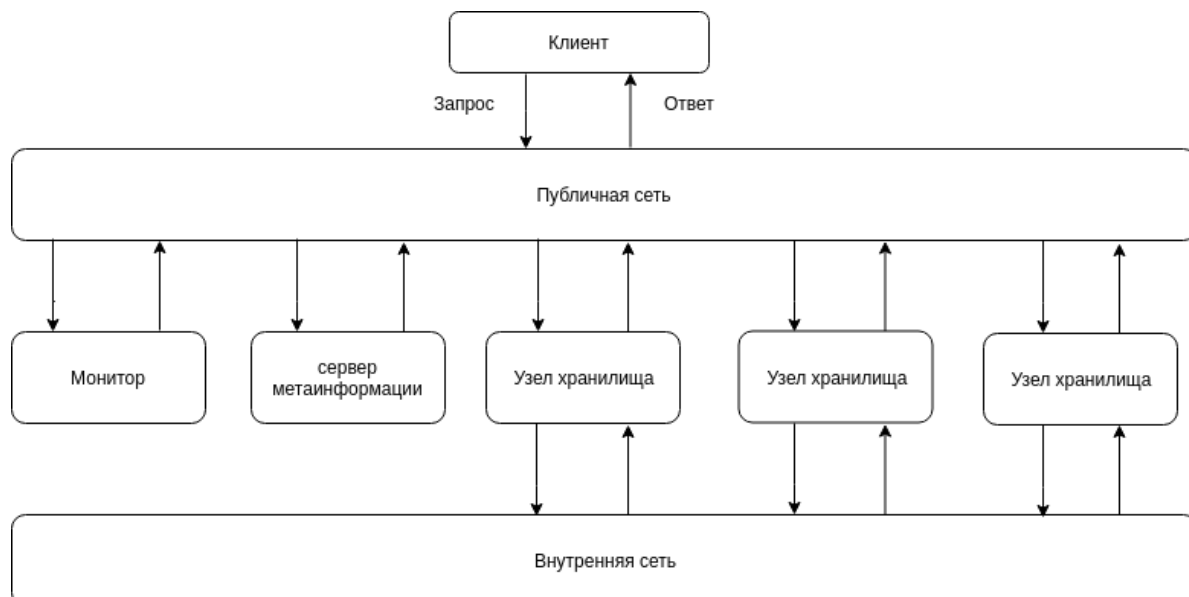


Рисунок 2 - Схема сетей файлового хранилища

Необходимо заменить информацию о внутренней и внешней сети в параметрах `public_network` и `cluster_network` соответственно. Параметр `monitor_interface` необходимо заменить на название сетевого интерфейса, используемого публичной сетью.

Проверить доступность узлов при помощи команды

```
$ ansible -i ../ceph_hosts all -m ping
```

Запуск развертывания распределенной файловой системы

```
$ cd ansible-master-distro
```

```
$ ansible-playbook site-docker.yml -i ../ceph_hosts
```

2.3.4 Получение данных для авторизации в системе хранения

Для получения данных, необходимо выполнить скрипт `get_key_fsid.sh` в домашней директории при помощи команды:

```
$ bashget_key_fsid.sh
```

Результатом выполнения команды является ответ вида:

```
CephClusterFSID: 21398661-64d4-4df9-9455-1c86bef1773c
```

```
CephClientKey: AQBcRxJdAAAAABAA3LPr+8IAXCCuvTeCnCoCLg==
```

Данные значения необходимо занести в файл `user-environment.yaml`, расположенный в домашней директории, в секцию `parameter_defaults`. Вид секции после занесения данных:

```
parameter_defaults:
CephClientKey: AQBcRxJdAAAAABAA3LPr+8IAXCCuvTeCnCoCLg==
CephClusterFSID: 21398661-64d4-4df9-9455-1c86bef1773c
CephExternalMonHost: 172.16.1.20, 172.16.1.21, 172.16.1.22,
172.16.1.23
```

где `CephExternalMonHost` содержит список IP-адресов узлов распределенной файловой системы.

2.3.5 Подготовка облачной среды

После настройки узла развёртывания в папке `/home/stack` появился файл `stackrc`, он должен быть активирован при каждой дальнейшей команде. Активация выполняется следующим образом

```
$ source stackrc
```

В файле `user-environment.yaml` по умолчанию указаны DNS сервер `8.8.8.8` и NTP сервер `pool.ntp.org`, для которых требуется доступ к интернету с машин, используемых для развёртывания облачной среды. При изолированном развёртывании необходимо поменять эти значения на другие существующие сервера, к которым должен быть доступ с узлов управления или вычислительных узлов.

Также необходимо в `user-environment.yaml` указать пароль для админского доступа к развёртываемой платформе в переменной `AdminPassword`.

2.3.6 Развертывание облачной среды

Регистрация узлов в instackenv.yaml:

Необходимо создать в директории /home/stack файл instackenv.json в формате, который описывает физические машины, используемые для развертывания облачной среды. Файл содержит JSON-объект с единственным ключом "nodes" и значением типа список. Список содержит JSON-объекты, каждый из которых соответствует одной машине. Для описания машины необходимо указать:

```
{
  "nodes": [
    {
      "name": <имя машины в базе данных undercloud, не влияет на саму машину>,
      "pm_type": "тип машины, доступные: ipmi, idrac, ilo",
      "pm_addr": <IP-адрес узла>,
      "pm_user": <имя пользователя для доступа к машине>,
      "pm_password": <пароль для доступа к машине>,
      "capabilities": "profile:<тип узла: compute или control>, node:<номер узла, пример: controller-0>",
      "ports": [
        {
          "address": <мас порта машины, по которому доступна загрузка по PXE>
        }
      ]
    },
    <...>
  ]
}
```

Если машина использует вместо BIOS - UEFI, то это необходимо отметить, добавив "boot_mode:uefi" к значению по ключу "capabilities"

Одновременно с интроспекцией узлов (флаги опциональны):

```
$ openstack overcloud node import --introspect --provide instackenv.yaml
```

Отдельно интроспекцию можно вызвать

```
$ openstack overcloud node introspect --all-manageable
```

Результаты интроспекции можно посмотреть следующим образом:

```
$ openstack baremetal introspection data save <NODE_ID>
```

Очистка узлов

```
$ openstack overcloud node clean --all-manageable
```

Предоставление узлов, перевод в состояние available

```
$ openstack baremetal node provide <NODE_NAME>
```

Note: Файлы необходимые для загрузки лежат в /var/lib/ironic/httpboot. При загрузке образов (ironic-python-agent), создаются два файла agent.ramdisk и agent.kernel

Если необходимо создать новый flavor такой же как дефолтные

```
$ openstack flavor create --disk 40 --ram 4096 openshift --property
resources:CUSTOM_BAREMETAL=1 --property resources:DISK_GB=0 --
property resources:MEMORY_MB=0 --property resources:VCPU=0 --
property capabilities:profile=openshift
```

Добавить профили для baremetal node (опционально; используется, чтобы разделить множество узлов на несколько; бесполезно, если используются scheduler hints):

```
$ openstack baremetal node show <NODE_NAME> -c properties
$ openstack baremetal node set <NODE_NAME> --property capabili-
ties=<prev value>,profile:control
```

Использовать scheduler hints, чтобы определить нумерацию узлов (опционально; используется для удобства):

```
$ openstack baremetal node set <NODE_NAME> --property capabili-
ties=<prev value>,node:compute=0
```

Для тех машин, у которых несколько дисков, обозначить root disk. Информацию о дисках можно посмотреть в результатах интроспекции:

```
$ openstack baremetal introspection data save <NODE_ID> | jq
.inventory.disks
$ openstack baremetal node set <NODE_NAME> --property
root_device='{ "serial": "6588a5a0e4f5370023903dddf178d222e" }'
```

Сгенерировать файл для работы с fencing:

```
$ openstack overcloud generate fencing -a reboot --ipmi-lanplus --
ipmi-level administrator ~/instackenv.yaml --output overcloud-heat-
templates/ispras-extra/fencing.yaml
```

Редактировать файлы templates/plan-environment.yaml и templates/network_data.yaml для удовлетворения собственным требованиям

Запустить развёртывание облака:

```
$ . deploy.sh overcloud
```

Убедиться, что развертывание успешно завершено.

```
Ansible passed.
Overcloud configuration completed.
Overcloud Endpoint: http://10.100.115.14:5000
Overcloud Horizon Dashboard URL: http://10.100.115.14:80/dashboard
Overcloud rc file: /home/stack/overcloudrc
Overcloud Deployed
+ status_code=0
+ openstack stack list
+ grep -q overcloud
+ openstack stack list
+ grep -Eq '(CREATE|UPDATE)_COMPLETE'
+ exit 0
```

Рисунок 3 - Результат успешного развертывания

Для доступа к развернутой облачной среде виртуальных машин необходимо узнать пароль для входа в графический интерфейс облачной среды при помощи команд

```
[stack@undercloud ~]$ source overcloudrc (overcloud)
[stack@undercloud ~]$ echo $OS_PASSWORD 3iNRjvVDPqaqM6TAdvSuPwpU2
```

Полученный пароль используется для авторизации в графическом интерфейсе облачной среды.

Первичная настройка облачной среды

Используя клиент командной строки:

Создание типа инстанса

```
$ openstack flavor create --ram 1024 --disk 10 --ephemeral 0 --swap
0 --vcpus 1 t2.micro
```

Загрузка образа VM

```
$ openstack image create --disk-format qcow2 --file cirros-0.4.0-
x86_64-disk.img cirros-0.4.0-x86_64
```

Создание ключевой пары

```
$ openstack keypair create --public-key .ssh/id_rsa.pub under-
cloud_key
```

Создание внутренней сети

```
$ openstack network create --external --provider-network-type
geneve --provider-segment 5555 internal
```

Создание внутренней подсети

```
$ openstack subnet create --dhcp --network internal --allocation-
pool start=192.168.33.5,end=192.168.33.250 --gateway auto --subnet-
range 192.168.33.0/24 internal_subnet
```

Создание роутера для внешней сети

```
$ openstack router create --external-gateway external router_1
$ openstack router add subnet router_1 internal_subnet
```

Создание правила группы безопасности для доступа к ВМ по ssh

```
$ openstack security group create ssh_access
$ openstack security group rule create --ingress --dst-port 22 --
protocol tcp --remote-ip 0.0.0.0/0 ssh_access
openstack security group rule create --ingress --protocol icmp --
remote-ip 0.0.0.0/0 ssh_access
```

Создание виртуальной машины с именем testing_config из образа

```
$ openstack server create --image cirros-0.4.0-x86_64 --flavor
t2.micro --security-group ssh_access --key-name undercloud_key --
network internal --wait testing_config
```

Назначение плавающего IP адреса виртуальной машине

```
$ openstack server add floating ip testing_config $(openstack
floating ip create external -c id -f value)
```

Добавление новых узлов

Для добавления нового узла, первым шагом необходимо зарегистрировать и выполнить интроспекцию новых узлов так же, как это производилось при первоначальном развёртывании облака.

Затем необходимо увеличить число вычислительных узлов в файле конфигурации (Параметр ComputeCount или ComputeInstanceNCount в случае развёртывания с поддержкой высокой доступности)

После чего необходимо запустить скрипт обновления развёртывания облака.

Замена вычислительных узлов

Прежде чем удалять вычислительный узел или узел хранения, необходимо убедиться, что он не используется.

Отключение вычислительного узла

Процесс отключения вычислительного узла означает перенос с него рабочей нагрузки для того, чтобы он мог быть отключен без ущерба для доступности вирту-

альных машин пользователей. Эта процедура может выполняться при перезагрузке вычислительного узла для обеспечения обновлений пакетов (например, после обновления ядра), при замене оборудования вычислительного узла, настройке ключей. Позволяет консолидировать рабочую нагрузку на меньшее количество машин при уменьшении переполнения;

Удаление определенных узлов из overcloud происходит с помощью команды: `openstack overcloud node delete --stack overcloud --templates [директория с шаблонами] <список идентификаторов>`

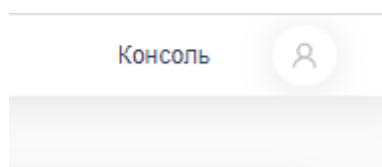
Для просмотра списка идентификаторов используется команда `openstack baremetal node list`

3. Эксплуатация облачной среды

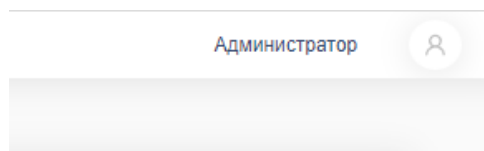
3.1 Доступные разделы веб-интерфейса

В панели управления облачной средой доступны два режима, которые далее называются «Страница в режиме консоли» и «Страница в режиме администратора». Если у вас нет административных прав, страница администратора будет недоступна, и вы будете автоматически попадать только в страницу консоли. Переход между этими режимами производится при помощи кнопки в правом верхнем углу:

Переход в режим "Консоль"

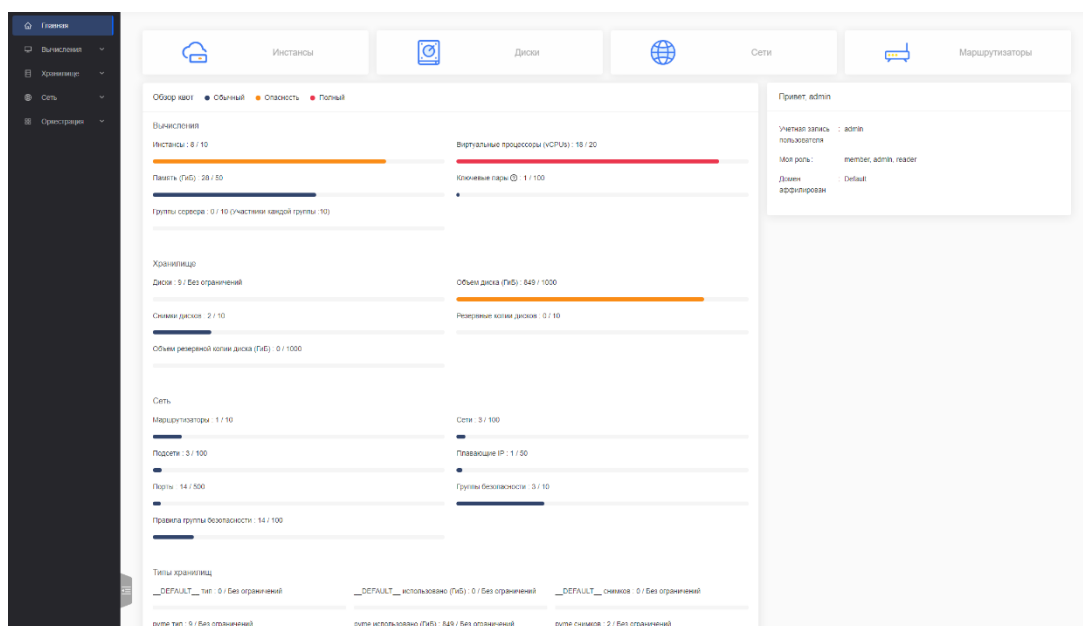


Переход в режим "Администратор"



3.2 Страница в режиме консоли

На вкладке «Главная» отображается информация о пользователе и детали квоты по Вычислениям, Хранению, Сети и т.д. Также предоставляется кнопка для быстрого перехода к страницам Вычислений, Хранилища, Сетей и Маршрутизаторов.



Вкладка «Вычисления»

Просмотр, запуск, удаление, создание инстансов (виртуальных машин), прикрепление или открепление интерфейсов, прикрепление или открепление тома, присвоение плавающего IP, управление группами безопасности, остановка, пауза, блокировка, шелвинг, приостановка, перезагрузка или мягкая перезагрузка экземпляров, изменение тегов экземпляра, или подключение к ним через VNC.

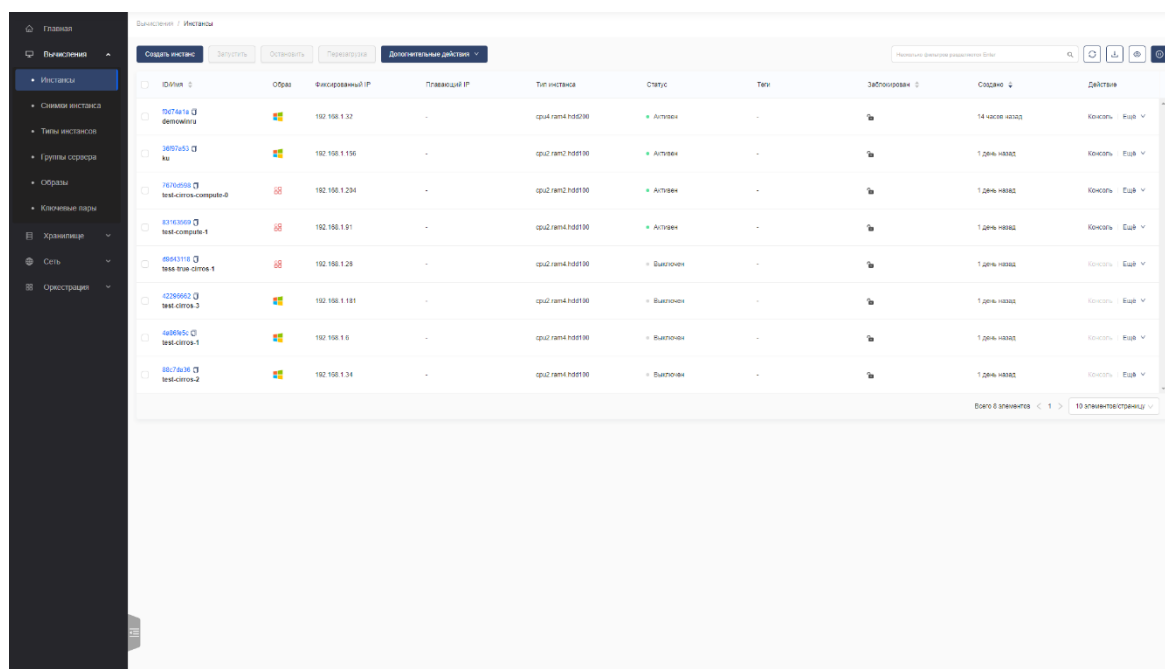
Просмотр, редактирование, удаление снимков экземпляров, запуск экземпляров или создание томов из них.

Просмотр конфигураций.

Просмотр, создание или удаление групп серверов.

Просмотр, создание, редактирование, удаление образов, запуск экземпляров или создание томов из них.

Просмотр, создание, редактирование, импорт и удаление пар ключей.

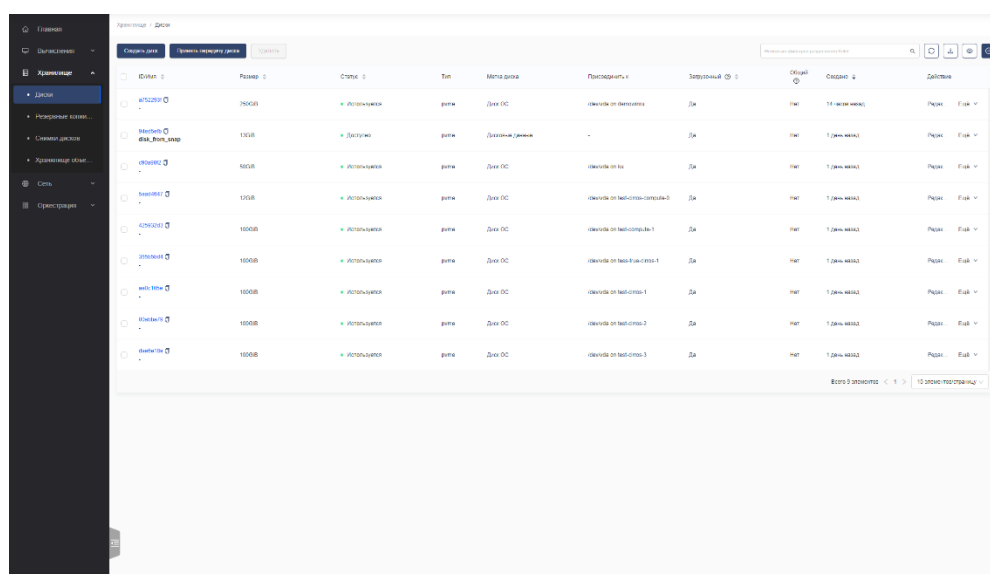


Вкладка «Хранилище»

Просмотр, создание, редактирование и удаление присоединяемых дисков. Создание снимков дисков, создание резервных копий дисков, клонирование дисков, расширение дискового пространства или изменение типа диска, прикрепление или открепление дисков от виртуальных машин.

Просмотр, создание, редактирование и удаление резервных копий дисков. Также создание дисков из них и восстановление резервной копии.

Просмотр, создание, редактирование, удаление снимков дисков, а также создание дисков из них.



Вкладка «Сеть»

Просмотр, создание, редактирование и удаление сетей.

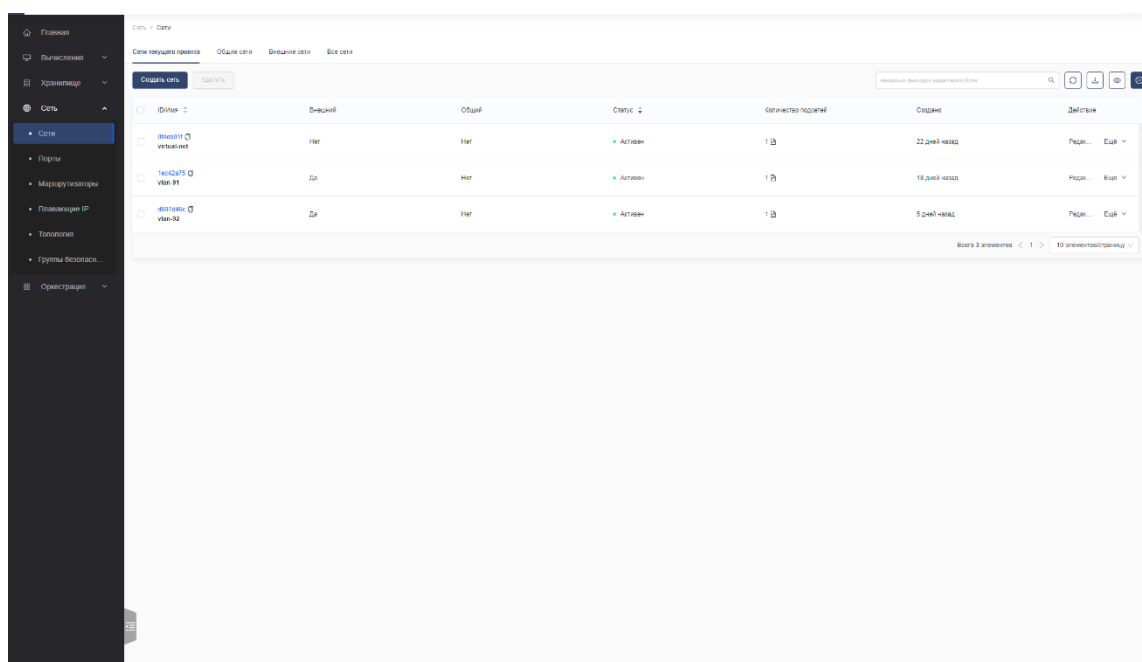
Просмотр, создание, редактирование, удаление портов и управление группой безопасности для портов.

Просмотр, создание, редактирование, удаление и управление маршрутизаторами.

Выделение IP-адресов или их освобождение.

Просмотр сетевой топологии.

Просмотр, создание, редактирование и удаление групп безопасности и правил групп безопасности.



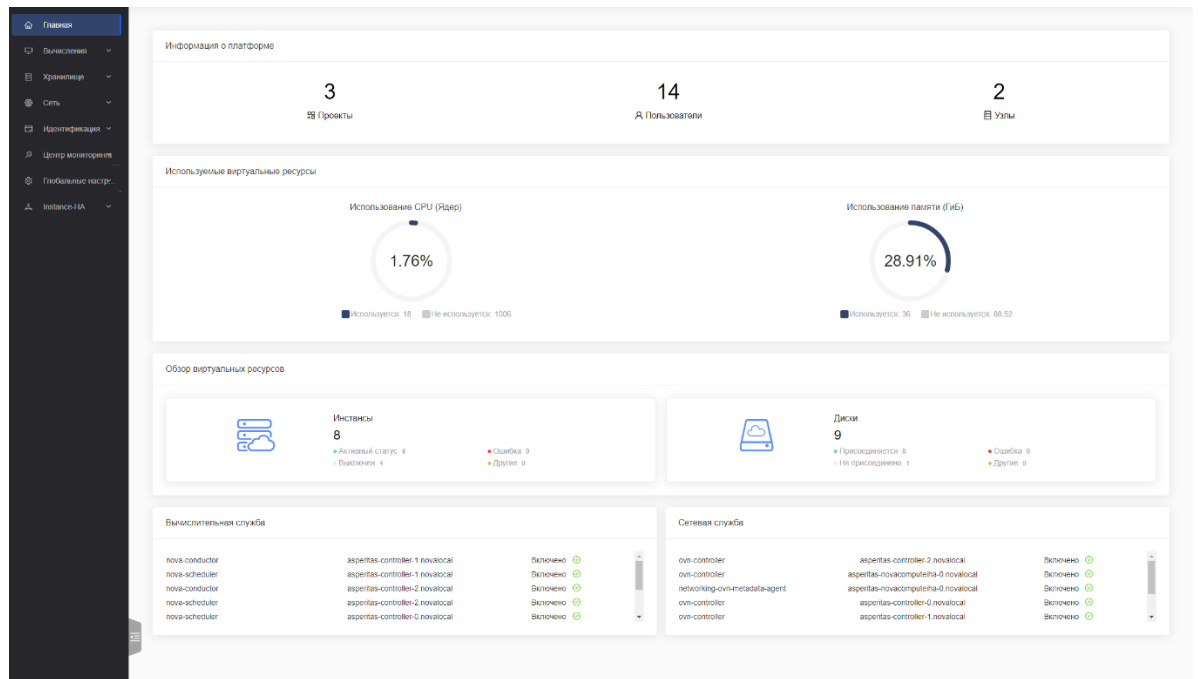
Вкладка «Оркестрация»

В данный момент не представляет особого интереса. Может быть полезна только в случае, если вы знакомы с системой Openstack Heat.

3.3 Страница в режиме администрирования

Вкладка «Главная»

На вкладке «Главная» отображается базовая информация о платформе (число проектов, пользователей, узлов), использование виртуальных ресурсов (использование CPU, использование памяти), статус вычислительных и сетевых услуг.



Вкладка «Вычисления»

Просмотр, остановка, приостановка, холодная или живая миграция, мягкая или жесткая перезагрузка и удаление экземпляров, принадлежащих всем проектам. Также просмотр журнала для экземпляра или доступ к экземпляру через VNC.

Просмотр, редактирование, удаление снимков экземпляров.

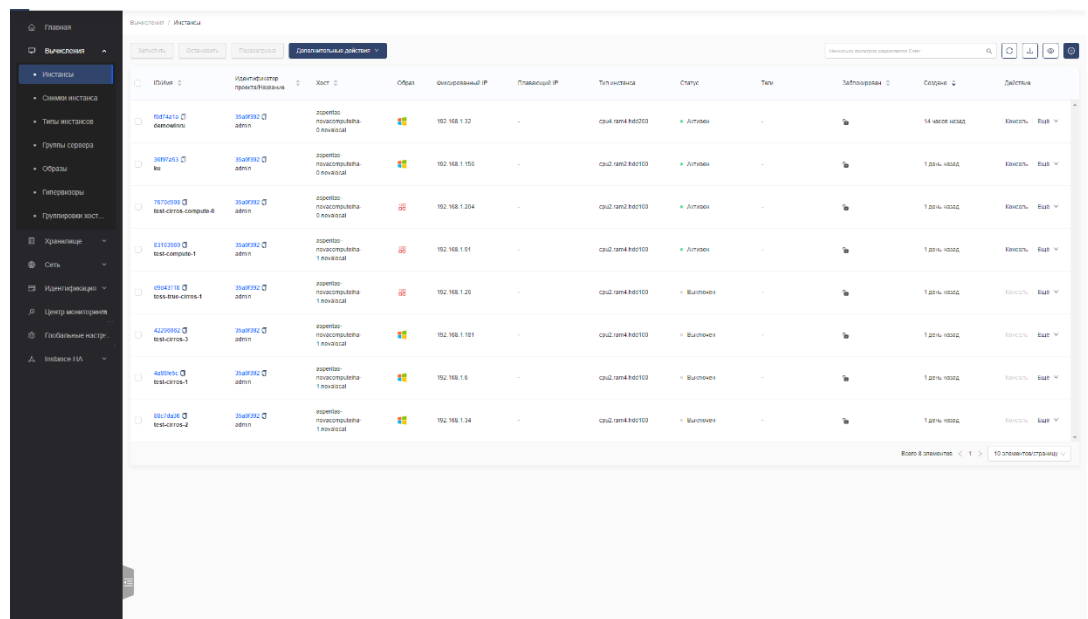
Просмотр, создание, редактирование, управление метаданными и удаление конфигураций.

Просмотр, создание или удаление групп серверов.

Просмотр, создание, редактирование, управление метаданными и удаление образов.

Просмотр сводки гипервизоров. Также просмотр и управление вычислительными узлами.

Просмотр, создание, управление метаданными, редактирование и удаление агрегатов хостов. Просмотр списка зон доступности.



Вкладка «Хранилище»

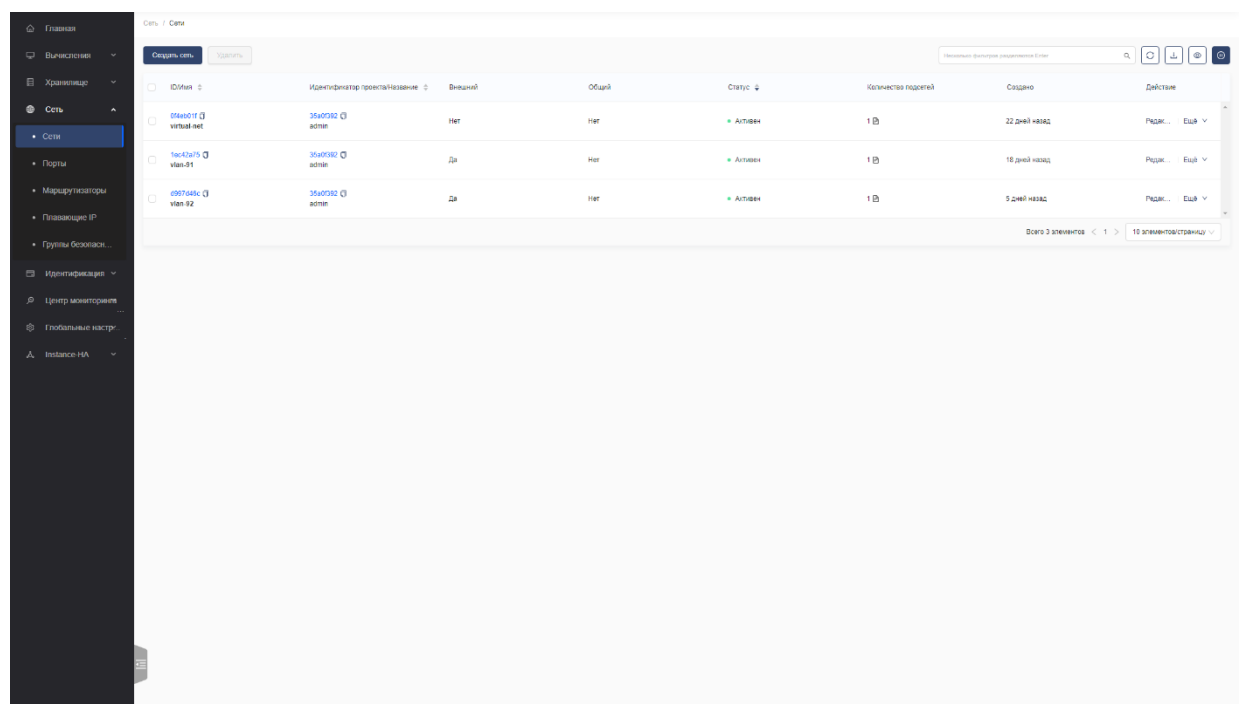
Просмотр, обновление статуса, миграция и удаление дисков.

Просмотр и удаление резервных копий дисков.

Просмотр и удаление снимков дисков.

Просмотр, создание, редактирование, шифрование, управление доступом и удаление типов дисков.

Просмотр состояния бэкендов хранения.



Вкладка «Сеть»

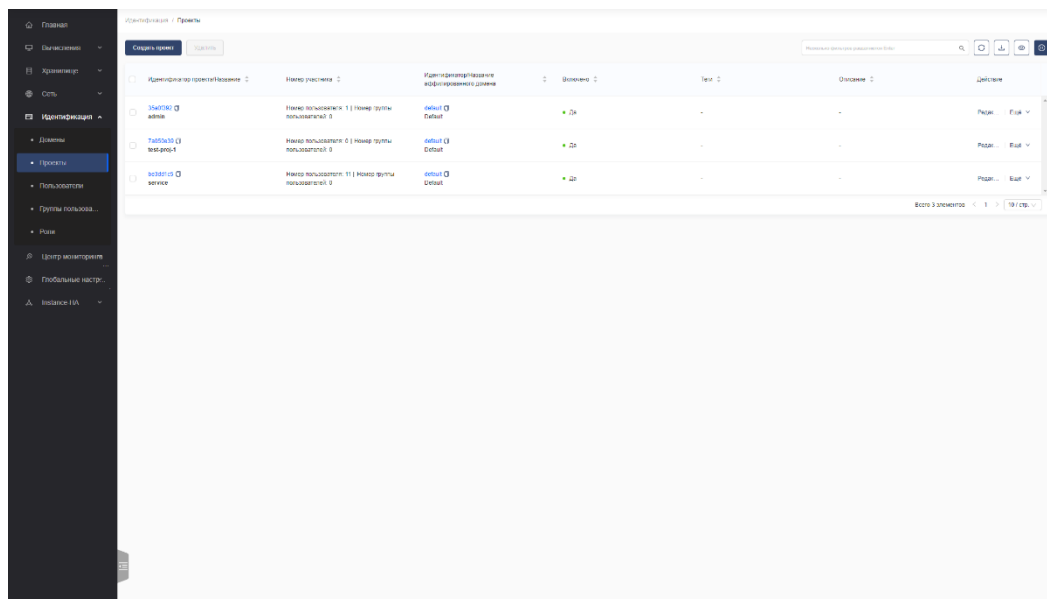
Просмотр, создание и удаление сетей.

Просмотр и удаление портов.

Просмотр и удаление маршрутизаторов.

Выделение IP-адресов или их освобождение.

Просмотр и удаление групп безопасности.



Вкладка «Идентификация»

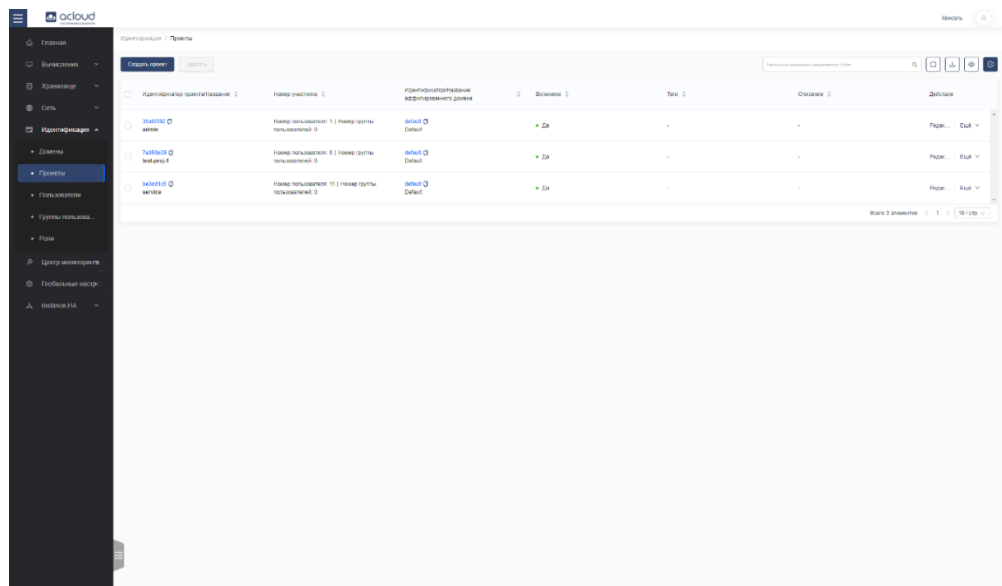
Просмотр, создание, редактирование, включение, отключение и удаление доменов.

Просмотр, создание, редактирование, включение, отключение и удаление проектов. Также управление пользователями или группами пользователей проектов, изменение тегов для них.

Просмотр, создание, редактирование, включение, отключение, удаление пользователей. И редактирование системного разрешения пользователей, обновление пароля пользователя.

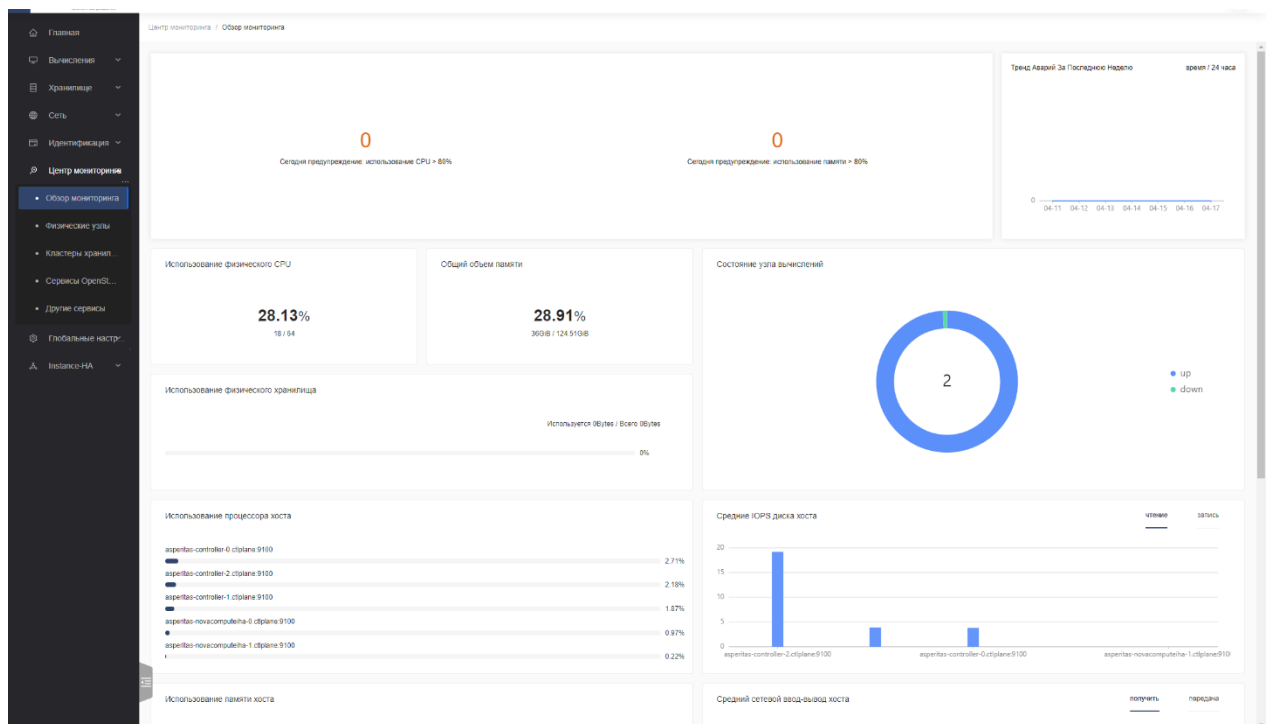
Просмотр, создание, редактирование и удаление групп пользователей.

Просмотр, создание, редактирование и удаление ролей.



Вкладка «Центр мониторинга»

В данном разделе сосредоточены упрощённые панели отображения метрик, собираемых мониторингом Асперитас.



Вкладка «Глобальные настройки»

Используйте следующие вкладки для просмотра информации об услугах:

Просмотр списка услуг.

Просмотр списка всех вычислительных услуг и включение или отключение их.

Просмотр сетевых агентов и включение или отключение их.

Просмотр списка всех услуг блочного хранения и включение или отключение их.

Просмотр списка всех точек доступа для инструментов автоматизации (ansible и тому подобных).

Просмотр, редактирование и сброс системной конфигурации.

Просмотр, редактирование и удаление системных метаданных.

Имя	Хост	Точка доступа	Статус сервиса	Состояние сервиса	Последнее обновление	Действия
nova-conductor	aspen-1-controller-1.novalocal	internal	Включено	Включено	1 минута назад	
nova-scheduler	aspen-1-controller-1.novalocal	internal	Включено	Включено	1 минута назад	
nova-conductor	aspen-2-controller-2.novalocal	internal	Включено	Включено	1 минута назад	
nova-scheduler	aspen-2-controller-2.novalocal	internal	Включено	Включено	1 минута назад	
nova-conductor	aspen-6-controller-6.novalocal	internal	Включено	Включено	1 минута назад	
nova-compute	aspen-6-nova-compute-6.novalocal	nova	Включено	Включено	1 минута назад	Отключить
nova-compute	aspen-1-nova-compute-1.novalocal	nova	Включено	Включено	1 минута назад	Отключить

3.4 Запуск и управление инстансами

3.4.1 Создание пары ключей

Для управления и входа в создаваемые инстансы можно использовать пароли (задаются на этапы создания инстанса) или ключевые пары (в системе используются ключи ssh; под ключевой парой подразумевается публичный и закрытый ключ из пары ssh-ключей). Ключевая пара может использоваться для входа *nix-like систему и для безопасной расшифровки пароля в Windows системах, если он был задан таким способом.

Войдите в дашборд.

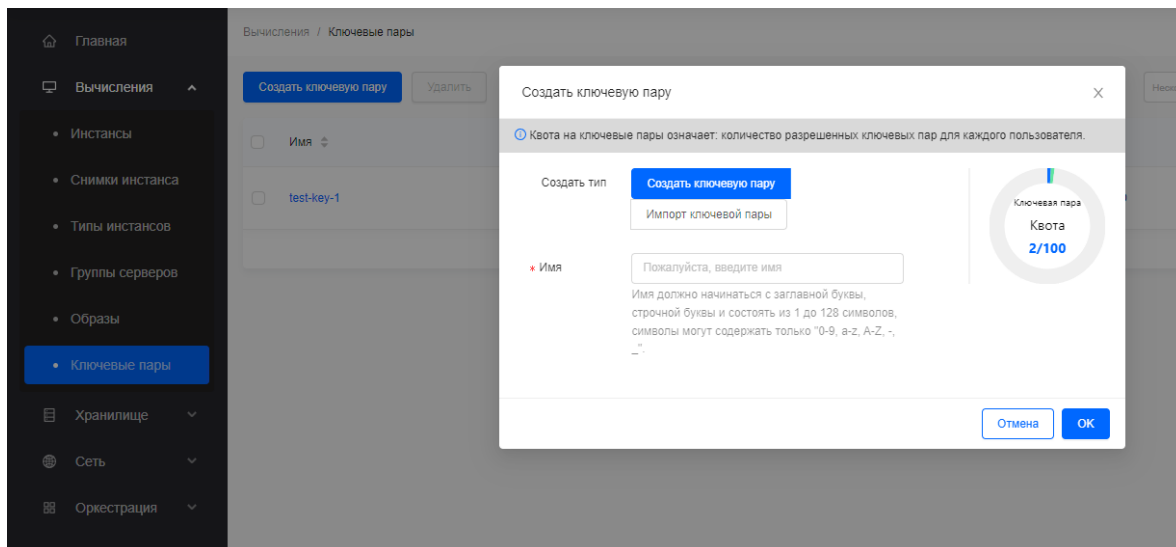
Выберите соответствующий проект из меню переключения проектов в верхнем левом углу.

admin ⇐ | Default

Вычисления / Инстансы

1. На странице **Консоль** откройте вкладку **Вычисления** и нажмите на катего-

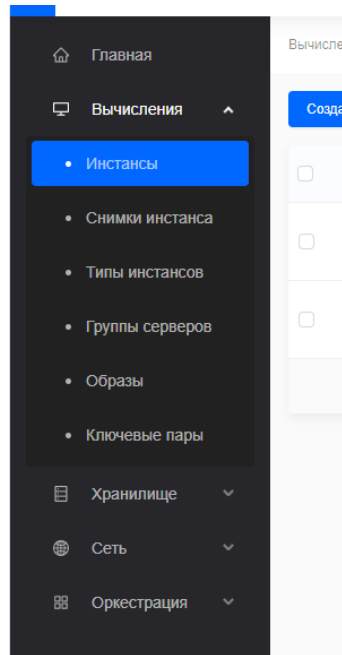
рию **Ключевые пары**.



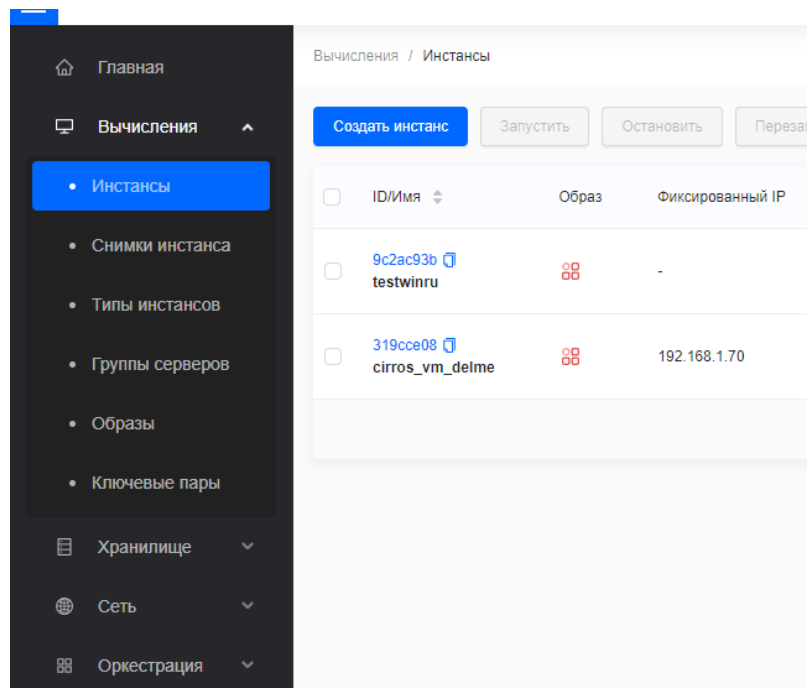
1. Нажмите **Создать пару ключей**.
2. В диалоговом окне **Создать ключевую пару** выберите один из вариантов **Создать тип**:
 - **Создать ключевую пару**: Если выберете этот вариант, введите **Имя**
 - **Импорт ключевой пары**: Если выберете этот вариант, появится новое поле для **Публичный ключ**. Введите **Имя** вашей ключевой пары, скопируйте публичный ключ в поле **Публичный ключ**.
3. Нажмите **ОК**. Дашборд отобразит ключевую пару на вкладке **Ключевые пары**.

3.4.2 Запуск инстанса

1. Войдите в дашборд.
2. Выберите соответствующий проект из меню **Switch Project** в верхнем левом углу.
3. На странице **Консоль** откройте вкладку **Вычисления** и нажмите на категорию **Инстансы**.

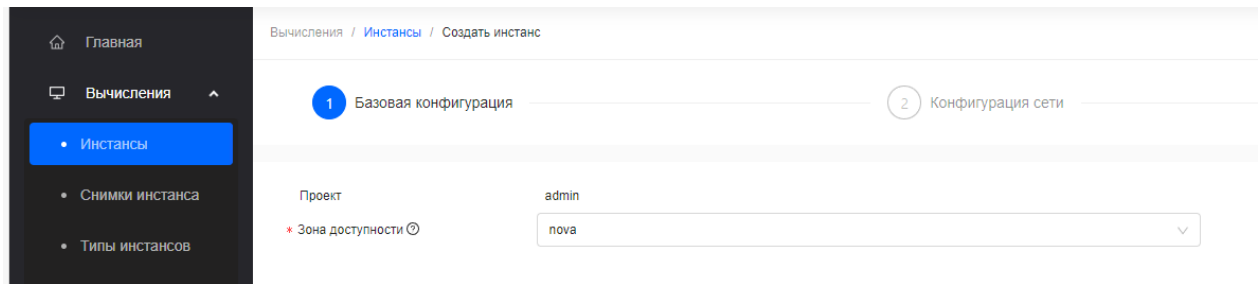


1. Нажмите **Создать инстанс**.

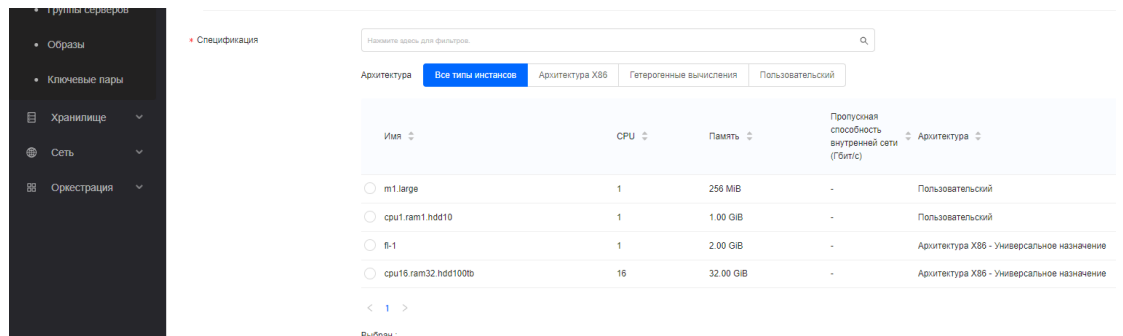


1. На странице **Создание инстанса** введите следующие значения для запускаемого инстанса.

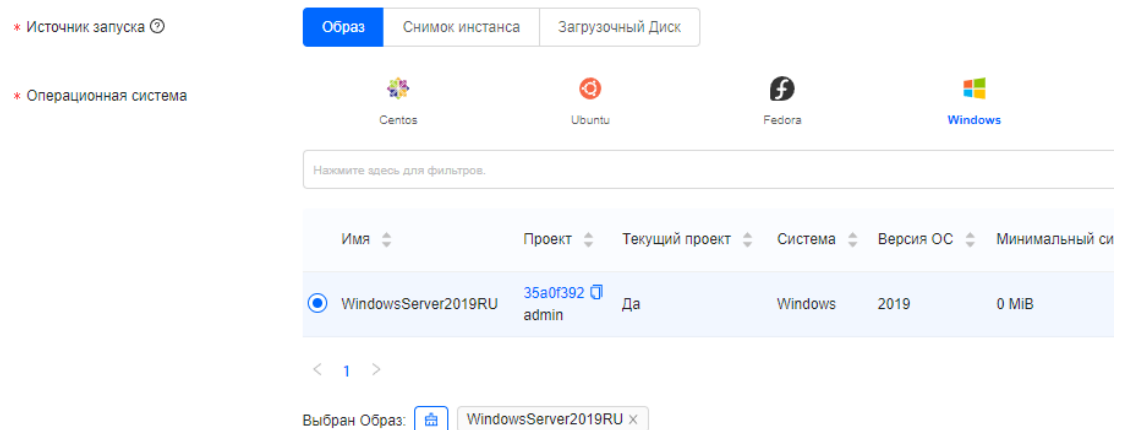
- **Зона доступности:** Зона доступности, в которой будет запущен сервер. В случае с облаками малого размера можно ничего не выбирать.



- **Спецификация:** Выберите конфигурацию для вашего инстанса, исходя из вычислительных потребностей.



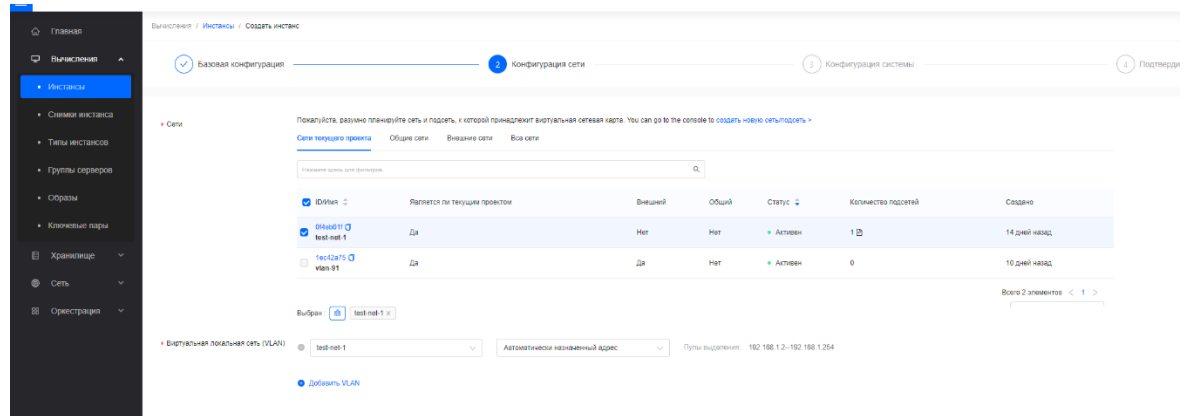
- **Источник запуска:** Выберите один из следующих вариантов:
 - **Образ:** Если выберете этот вариант, появится новое поле для **Операционная система**. Вы можете выбрать образ из списка. И введете размер диска, используемого как **Системный диск** инстанса.



- **Снимок инстанса:** Используя этот вариант, вы можете загрузиться с снимка диска и создать новый диск, выбрав **Снимок инстанса** из списка.
 - **Загрузочный диск:** Если выберете этот вариант, появится новое поле для **Загрузочный диск**. Вы можете выбрать диск из списка.
 - **Данные диска:** Диски, подключенные к инстансу.
2. Нажмите **Далее: Конфигурация сети**

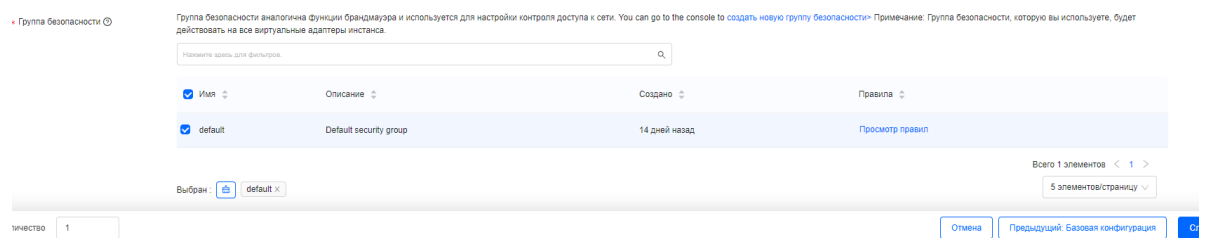
Вы можете выбрать Сети, Порты или их комбинацию для сетевой конфигурации инстанса.

- **Сети:** Добавьте сеть к инстансу. Если вы указываете сети, Виртуальная локальная сеть (VLAN) и Группа безопасности являются обязательными полями.



- **Виртуальная локальная сеть (VLAN):** Укажите подсеть сети и назначьте фиксированный IP адрес автоматически или вручную для инстанса.

Группа безопасности: Группы безопасности - это вид облачного файервола, который определяет, какой входящий сетевой трафик перенаправляется на инстансы.



Порты: Активируйте порты, которые вы хотите назначить инстансу.

3. Нажмите **Далее: Конфигурация системы**.

Имя: Имя сервера.

Тип входа в систему: Выберите один из следующих вариантов:

- **Пара ключей:** Если выберете этот вариант, появится новое поле для Пара ключей. Пара ключей позволяет вам подключиться по SSH к вашему вновь созданному инстансу. Вы можете выбрать существующую пару ключей, импортировать пару ключей или создать новую пару ключей.
- **Пароль:** Введите Пароль для входа и подтвердите его. И вы можете войти в инстанс, используя пароль.

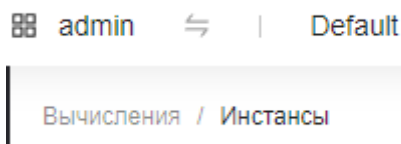
4. Нажмите **Далее: Подтверждение конфигурации** и подтвердите свой выбор.

Инстанс создается и вы можете подождать несколько секунд, чтобы следить за изменениями списка данных инстанса или вручную обновить данные, чтобы получить окончательный результат отображения.

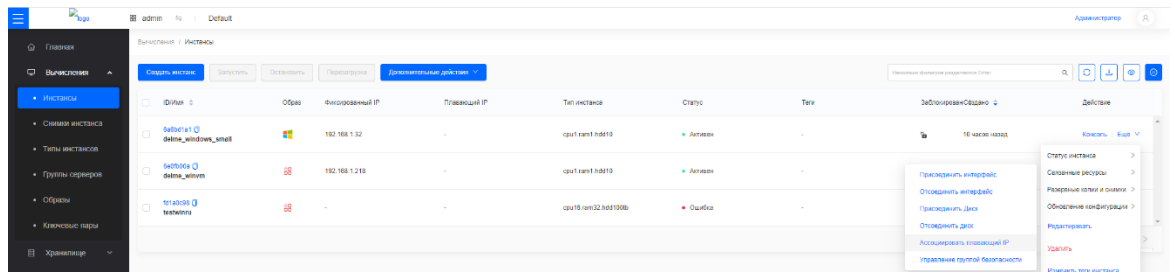
3.4.3 Назначение плавающего IP-адреса инстансу

Плавающие IP — это очень важная концепция в облачной среде. В рамках облачной среды зачастую используются сети, которые являются локальными для данного проекта в облаке. Они зачастую строятся на технологиях VLAN или оверлейных сетях (например, geneve или vxlan). С другой стороны, для части серверов важно быть видимыми из интернета или из внутренних сетей на стороне организации. В этом случае зачастую используются *плавающие IP*: они привязываются к инстансам по мере необходимости (и их можно отвязать и переназначить другим инстансам). Сама операционная система инстанса про такие IP ничего не знает; они срабатывают как NAT-адреса для доступа к инстансу.

1. Войдите в дашборд.
2. Выберите соответствующий проект из меню переключения проектов в верхнем левом углу.



1. На странице **Консоль** откройте вкладку **Вычисления** и нажмите на категорию **Инстансы**.
2. В столбце **Действие** выберите **Связанные ресурсы** и нажмите **Назначить плавающий IP**.



1. В диалоговом окне **Назначение плавающего IP** выберите **IP инстанса** и **Плавающий IP-адрес**.
2. Нажмите **ОК**.

Важное замечание

Чтобы отменить назначение IP-адреса инстансу, нажмите кнопку **Отменить назначение плавающего IP**.

3.4.4 Загрузка образа

Образы используются для создания виртуальных машинных инстансов в облаке.

1. Войдите в дашборд.
2. Выберите соответствующий проект из меню **Switch Project** в верхнем левом углу.
3. На странице **Консоль** откройте вкладку **Вычисления** и нажмите на категорию **Образы**.
4. Нажмите **Создать образ**.
5. На странице **Создание образа** введите следующие значения:
Имя: Имя образа.

Тип загрузки: Выберите один из следующих вариантов:

- **Загрузить файл:** Если выберете этот вариант, нажмите **Нажмите для загрузки** чтобы загрузить двоичный файл данных образа.
- **URL файла:** Если выберете этот вариант, введите **URL файла**.

Формат диска: Выберите формат образа (например, QCOW2) для образа.

Формат контейнера: Для виртуальных машин выбирайте **Варг**. Для образов Docker, выбирайте **Docker** (но это нужно, только если вы точно знаете, что в системе есть узлы, использующие контейнеры вместо виртуальных машин, иначе в этом нет смысла).

Операционная система: Выберите операционную систему образа (например, CentOS).

Версия ОС: Версия операционной системы образа. Используйте общепринятую нотацию; система обращает внимание на номера версий.

Администратор ОС: Имя администратора операционной системы образа. В общем случае, Administrator для Windows, root для Linux.

Мин. системный диск (ГБ): Количество дискового пространства в ГБ, необходимое для загрузки образа.

Мин. память (ГБ): Количество памяти в ГБ, необходимое для загрузки образа.

Защищенный: Защита образа от удаления.

Тип использования: Выберите тип использования (например, Общий сервер) для образа.

Описание: Человеко-читаемое описание ресурса.

Дополнительные опции: Имеет смысл менять, только если вы точно знаете, что делаете.

The screenshot shows the 'Создать образ' (Create Image) form in the Yandex Cloud console. The left sidebar contains navigation links: Главная, Вычисления, Хранилище, Сеть, and Оркестрация. The 'Вычисления' section is expanded, showing options like Инстансы, Шаблоны инстансов, Типы инстансов, Группы серверов, and Образы (which is selected). The main form area is titled 'admin / Default' and 'Вычисления / Образы / Создать образ'. It includes fields for:

- Имя:** A text field with a placeholder 'Пожалуйста, введите имя' and a note that the name must start with a letter, contain only letters, numbers, and hyphens, and be 1-128 characters long. There are buttons for 'Загрузить файл' and 'URL файла'.
- Тип загрузки:** A dropdown menu with 'File' selected. Below it is a button 'Нажмите для загрузки'.
- Формат диска:** A dropdown menu with 'Пожалуйста, выберите'.
- Формат контейнера:** A dropdown menu with 'Bare'.
- Операционная система:** A dropdown menu with 'Пожалуйста, выберите'.
- Версия ОС:** A text field.
- Администратор ОС:** A text field with a note: 'Обычно Administrator для Windows, root для Linux, запустите при загрузке образа.'
- Минимальный системный диск (ГБ):** A text input field.
- Минимальная память (ГБ):** A text input field.
- Защищенный:** A checkbox labeled 'Защищенный'.
- Тип использования:** A dropdown menu with 'Общий сервер' selected.
- Описание:** A large text area.
- Дополнительные опции:** A link 'Развернуть дополнительные параметры'.

6. Нажмите **Подтвердить**.

Образ ставится в очередь на загрузку. Может потребоваться некоторое время, прежде чем статус изменится с **В очереди** на **Активный**.

4. Описание процессов, обеспечивающих поддержание жизненного цикла ПО

4.1 Процессы разработки и совершенствования ПО

Разработка инструментов развертывания облачной среды Асперитас ведется по методологии Agile с привлечением современных средств повышения качества кода.

1. Для хранения кода используется система контроля версий git, и изменения в основной ветке проходят инспекцию кода (code review) другими разработчиками.
2. При написании кода разработчики должны придерживаться строгих правил оформления кода; нарушение этих правил не позволит пройти этап инспекции кода.
3. Используется инструментарий, позволяющий выполнять первичную валидацию конфигурации.
4. Периодически производится проверка кода среды на предмет наличия ошибок

4.2 Поддержка пользователей ПО

Пользователи, которым достаточно для решения их задач возможностей уже реализованных в среде анализа алгоритмов, могут приступить к работе со средой после краткого обучения, проводимого на стороне разработчика (ИСП РАН). В ходе своей работы среда Асперитас автоматически ведёт журналы действий и ошибок, которые могут быть отправлены разработчику в случае обнаружения некорректного поведения или возникновения запроса на улучшение и доработку.

Информация о сбоях в работе среды Асперитас, проблемах производительности, ошибках целевого функционала передаются пользователями среды непосредственно ответственным сотрудникам ИСП РАН, без использования публичных Интернет-ресурсов управления ошибками. Это обеспечивает должный уровень конфиденциальности для контрольных примеров, передаваемых пользователем среды в ИСП РАН для оценки и исправления программного дефекта.

Со своей стороны ИСП РАН постоянно совершенствует Асперитас, применяя в жизненном цикле разработки передовые методики. Добавление новых и улучшение существующих алгоритмов ведется в инициативном порядке. Обновления среды Асперитас передаются пользователям среды через согласованные с ними каналы распространения обновлений.

4.3 Необходимый персонал для разработки и поддержки

Персонал должен обладать следующими навыками:

1. Опыт работы и понимание принципов работы облачных систем.
2. Опыт инсталляции, настройки и администрирования пользовательского, системного и СПО.
3. Навыки и опыт администрирования ОС на базе ядра Linux.
4. Опыт построения кластерных систем.
5. Опыт работы и понимание архитектуры распределенных файловых систем.

Для разработки и поддержки программного продукта необходима соответствующая квалификация разработчиков. Это вызвано следующими причинами:

1. Специфическая предметная область, требующая глубоких знаний одновременно в нескольких областях: устройство современной аппаратуры и операционных систем, системное администрирование, компьютерная безопасность, технологии разработки ПО.
2. Наличие среди алгоритмов, реализованных в среде развертывания, алгоритмов, впервые разработанных сотрудниками ИСП РАН.
3. Требования к производительности системы.

В силу приведенных причин коллектив разработчиков Асперитас формируется из специалистов, получивших профильное образование.

Для гарантийного обслуживания задействовано 5 программистов и научных сотрудников, для Технической поддержки задействованы 3 программиста, для модернизации программного обеспечения задействованы 10 научных сотрудников и программистов.

Адрес электронной почты, по которому можно обратиться по вопросам, связанным с облачной средой «Асперитас» — asperitas@ispras.ru.