

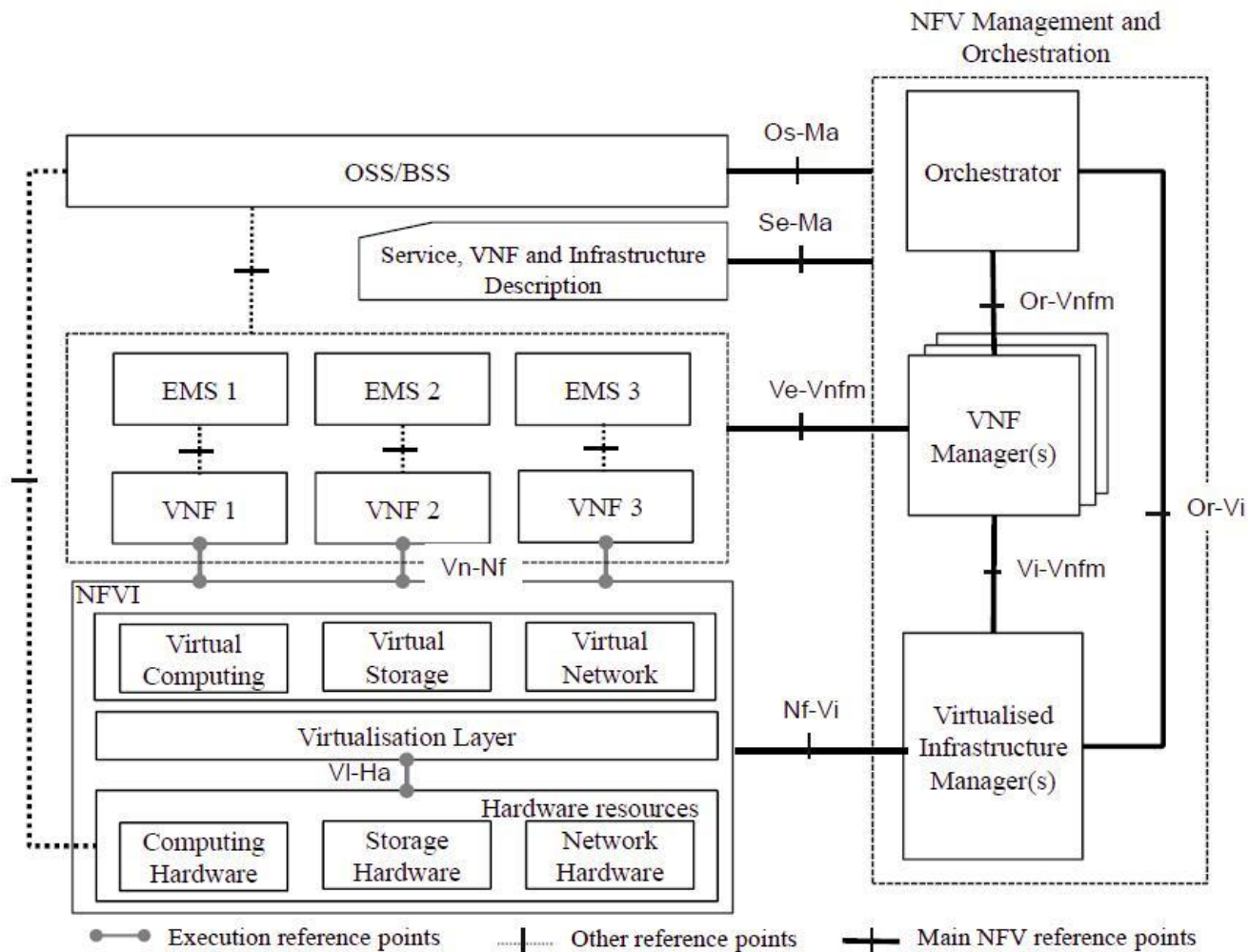
# Разработка системы обеспечения надежного и масштабируемого виртуального сетевого сервиса в облачной среде

Автор: Романов Андрей Романович

Научный руководитель: Антоненко Виталий Александрович

# Проблемы организации сетевых сервисов в традиционных сетях

- расчет производительности сервиса исходя из максимально возможной нагрузки
- простаивание оборудования при не пиковых нагрузках
- использование оборудования с излишней функции
- зависимость от производителя оборудования, на котором работает сервис



# Цели обзора

- исследовать достоинства и недостатки существующих решений
- сформировать требования к разрабатываемому решению

# Результаты обзоров

Платформа	ETSI NFV MANO	Независимость от платформы виртуализации	Поддержка нескольких VIM	Мониторинг состояния сервиса	Автоматическое срабатывание триггеров мониторинга
OPNFV	да	да	нет	не заявлено	не заявлено
Cloudify	да	да	не заявлено	да	нет
Openstack Tacker	да	нет	нет	не заявлено	не заявлено
OpenBaton	да	да	да	да	нет

# Постановка задачи

Разработать модуль управления виртуальными сетевыми сервисами.

Решение должно удовлетворять следующим свойствам:

1. Управление подпиской на сервис
2. Обеспечивать автоматическую масштабируемость сервиса
3. Обеспечивать отказоустойчивость сервиса
4. Архитектура должна соответствовать стандарту ETSI NFV MANO
5. Независимость от платформы виртуализации ресурсов
6. Поддержка нескольких платформ виртуализации ресурсов

# Особенности реализации

- Описания виртуальных сетевых функций являются входными данными для модуля
- Согласно архитектуре ETSI NFV MANO модуль выполняет функции оркестратора
- Модуль интегрирован в облачную платформу Cloud Conductor



# Эксперименты

В целях проверки автоматического срабатывания триггеров было проведено два эксперимента healing и scaling.

Healing - сервис перестает работать. Задача модуля - восстановить работу сервиса.

Scaling - обнаружение нехватки ресурсов при работе сервиса. Задача модуля заключается в размещении дополнительной инфраструктуры и проверке, что теперь сервис работает в штатном режиме.

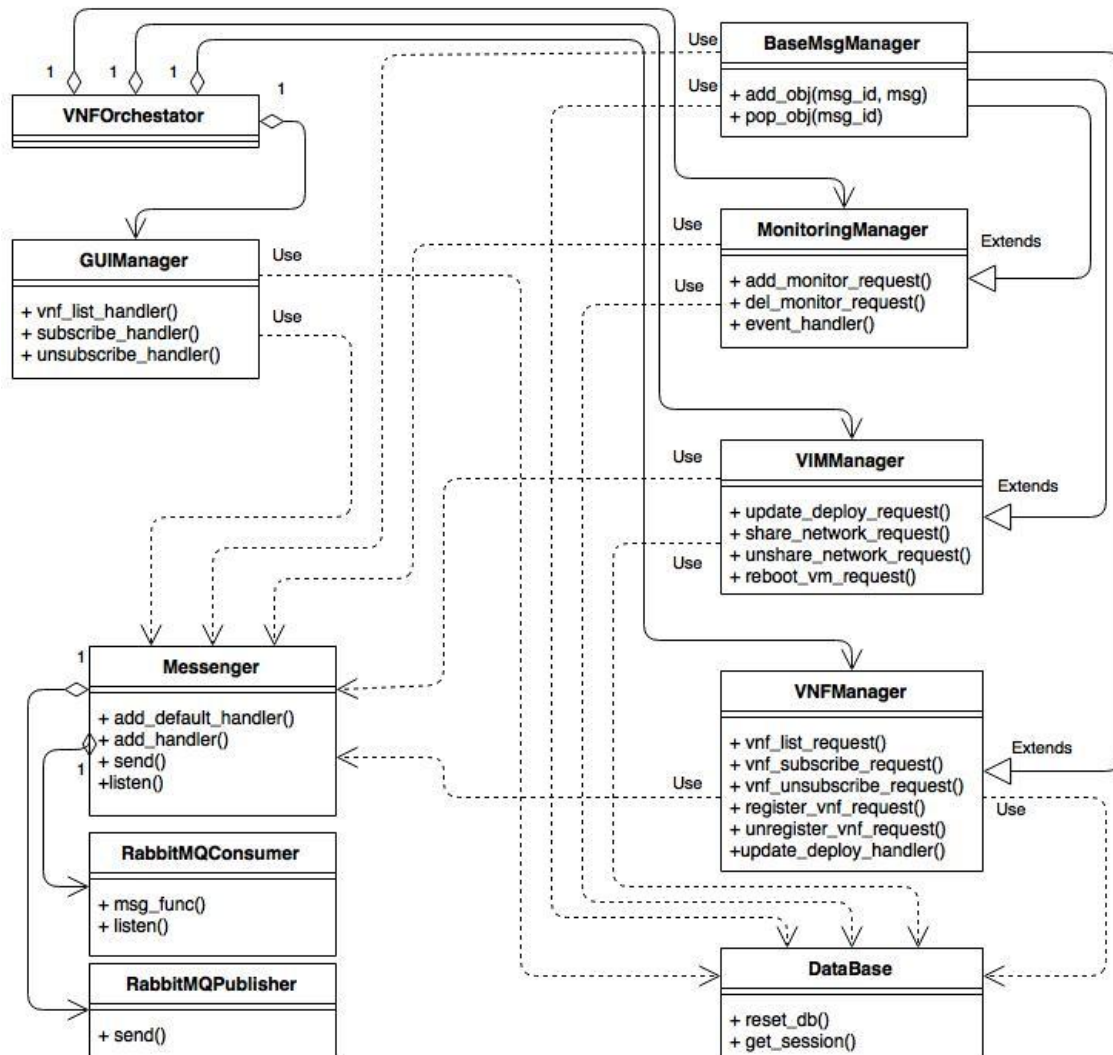


# Результаты

- Разработан модуль, интегрированный в платформу Cloud Conductor, управляющий виртуальными сетевыми сервисами и обеспечивающий их масштабируемость и отказоустойчивость
- Архитектура разработанного модуля согласована с высокоуровневым стандартом ETSI NFV MANO

Вопросы?





# Описания виртуальных сетевых функций

Описание должно содержать:

- параметры функции, вводимые пользователем
- топологию инфраструктуры, необходимой для работы экземпляра функции
- схему запуска функции (скрипты и пр.)
- описания событий о сбоях в работе функции
- перечисление реакций на такие события
- другие параметры

# Реализация экспериментов

В качестве виртуальной сетевой функции предлагается взять прокси “squid”. Кроме предоставления прокси, машине будет предоставлен доступ в Интернет.

Услуга доступна  $\Leftrightarrow$  у машины есть доступ в Интернет

Масштабирование:

расширение инфраструктуры - подключение новых клиентов к услуге

сужение инфраструктуры - отключение клиентов от услуги

# Пример подписки на функцию (создание тенанта)

← Back   Deploy   Save   Delete   Save As Tmpl   Import   Export   Terminal

Subscribe to VNF

## TENANT MAP

+ Add Element   Add Link

```
graph TD; vm((vm)) --- network((network));
```

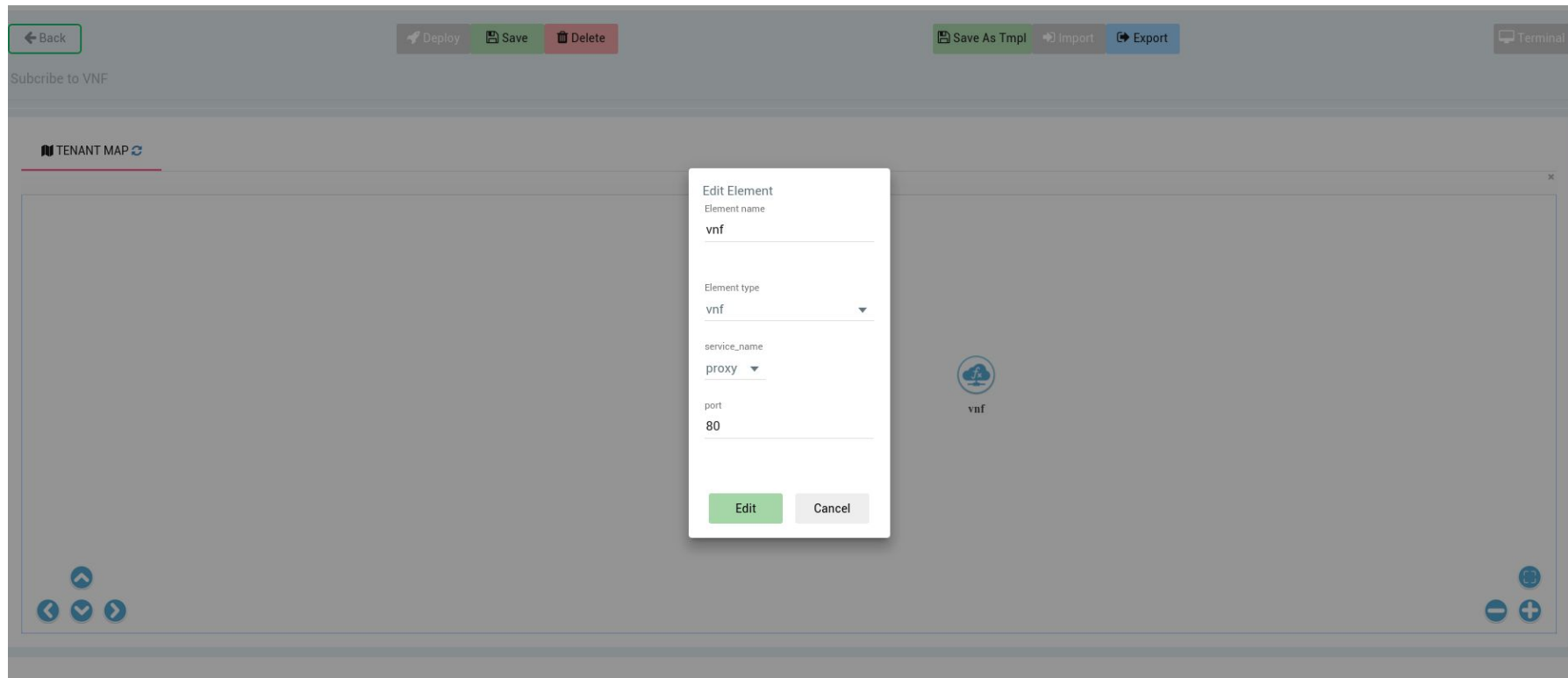
vm

network

Successfully updated tenant "undefined" in project "aromanov"

Request tenant + "dyplom" in project "aromanov" update

# Пример подписки на функцию (добавление vnf)



# Пример подписки на функцию (сохранение и деплой)

← Back   Deploy   Save   Delete   Save As Tmpl   Import   Export   Terminal

Subscribe to VNF

TENANT MAP ↻

Edit

```
graph TD; vnf((vnf)) -.-> network((L3)); vm((vm)) --- network;
```

vm   network

Successfully updated tenant "undefined" in project "aromanov"

Request tenant + "dyplo" in project "aromanov" update



# Пример подписки на функцию (подписка)

Subscribe to VNF

← Back   Undeploy   Save   Delete   Save As Tmpl   Import   Export   Terminal

TENANT MAP   Edit

Context Element

Subscribe VNF   Unsubscribe VNF   Check Result   Cancel

vm   vnf   network

Successfully deployed tenant "dyplom" in project "aromanov"

The screenshot displays a cloud management interface. At the top, there's a header bar with a 'Subscribe to VNF' title and a toolbar containing buttons for 'Back', 'Undeploy', 'Save', 'Delete', 'Save As Tmpl', 'Import', 'Export', and 'Terminal'. Below the header, the main area is titled 'TENANT MAP' with an 'Edit' button. A 'Context Element' dialog box is open in the center, featuring four buttons: 'Subscribe VNF' (green), 'Unsubscribe VNF' (blue), 'Check Result' (light blue), and 'Cancel' (red). The background shows a diagram with three nodes: 'vm' (a computer icon), 'vnf' (a cloud with a gear icon), and 'network' (a cloud with a plus icon). A dashed arrow points from 'vnf' to 'vm', and a solid line connects 'vm' to 'network'. In the bottom right corner, a green notification box states 'Successfully deployed tenant "dyplom" in project "aromanov"'. Navigation arrows are visible in the bottom left corner.

# Пример подписки на функцию (подписка 2)

Во время подписки прототип:

- 1) размещает дополнительную инфраструктуру под функцию
- 2) инициализирует инфраструктуру
- 3) подключает машину пользователя (в примере “vm”) к инфраструктуре функции.
- 4) сообщает об удачном завершении подписки

После этого клиент может начать использовать услугу, предоставляемую функцией.