

ОСНОВНЫЕ ТЕНДЕНЦИИ АППАРАТНОЙ И ПРОГРАММНОЙ ОРГАНИЗАЦИИ СОВРЕМЕННЫХ КОМПЬЮТЕРНЫХ СЕТЕЙ: ОТ СПЕЦИАЛИЗИРОВАННЫХ АППАРАТНЫХ ПЛАТФОРМ К ПРОГРАММНО-КОМПОЗИРУЕМЫМ СРЕДАМ

Ю. В. Савицкий¹, Г. Л. Муравьев²

¹ К. т. н., доцент, доцент кафедры интеллектуальных информационных технологий, УО «Брестский государственный технический университет», Брест, Беларусь, e-mail: yury.savitsky@tut.by

² К. т. н., доцент, доцент кафедры интеллектуальных информационных технологий, УО «Брестский государственный технический университет», Брест, Беларусь, e-mail: mgl_work@mail.ru

Реферат

В настоящее время современные компьютерные сети претерпевают фундаментальную реорганизацию, определяемую требованиями облачных вычислений (Cloud Computing), интернета вещей (IoT, Internet of Things), больших данных (Big Data) и необходимости чрезвычайно высокой гибкости и адаптивности. Данная статья посвящена исследованию и всестороннему анализу методов и средств аппаратной и программной организации сетей, переживающих эпоху глубокой декомпозиции и конвергенции. В работе исследуется эволюция аппаратных платформ от закрытых, специализированных интегральных схем (ASIC) вендор-специфичного оборудования (функционирующего оптимально только в рамках собственной экосистемы производителя – вендора) к открытым, программируемым архитектурам, сетевым процессорам (SmartNIC) и процессорам для обработки данных (DPU), что стирает границу между сетевым и вычислительным узлом. Также анализируется трансформация сетевого ПО: от монолитных операционных систем (являющихся частной собственностью разработчика, имеющих закрытый исходный код, а также ограничения на использование, модификацию и распространение) к парадигмам программно-определяемых сетей (SDN) и виртуализации сетевых функций (NFV).

В статье систематизированы современные средства организации сетей, включая языки программирования плоскостей данных (data plane programming languages), такие как P4. Основной вывод исследования заключается в том, что современная организация сети характеризуется переходом от жестко связанных, вертикально интегрированных аппаратно-программных архитектур к горизонтальным, программно-композируемым и децентрализованным моделям. В таких моделях сетевые сервисы и функции динамически развертываются на множестве гетерогенных вычислительных ресурсов под управлением интеллектуальных контроллеров и оркестраторов (централизованных инструментов, автоматизирующих управление, координацию, развертывание и масштабирование распределенных контейнерных приложений). Предлагаются перспективные направления будущих исследований, включая интеграционные платформы и средства комплексной организации сетей, в частности гиперконвергентные инфраструктуры; облачные нативные сетевые плагины для контейнерных оркестраторов; облачные сетевые сервисы.

Ключевые слова: аппаратная организация сетей, программная организация сетей, SDN, NFV, P4, SmartNIC, DPU, гиперконвергентная инфраструктура (HCI), IBN, декомпозиция, облачные нативные сети, оркестратор.

KEY TRENDS IN THE HARDWARE AND SOFTWARE ORGANIZATION OF MODERN COMPUTER NETWORKS: FROM SPECIALIZED HARDWARE PLATFORMS TO SOFTWARE-COMPOSABLE ENVIRONMENTS

Y. V. Savitsky, G. L. Muravov

Abstract

Modern computer networks are currently undergoing a fundamental reorganization driven by the demands of cloud computing, the Internet of Things (IoT), big data, and the need for extreme flexibility and adaptability. This article explores and comprehensively analyzes the methods and tools for hardware and software network organization in an era of deep decomposition and convergence. The paper examines the evolution of hardware platforms from closed, application-specific integrated circuits (ASICs) of vendor-specific hardware (operating optimally only within the manufacturer's ecosystem) to open, programmable architectures, network processors (SmartNICs), and data processing units (DPUs), blurring the line between network and computing nodes. The paper also analyzes the transformation of network software: from monolithic operating systems (proprietary, closed-source, and subject to restrictions on use, modification, and distribution) to software-defined networking (SDN) and network functions virtualization (NFV) paradigms.

The paper systematizes modern networking tools, including data-plane programming languages such as P4. The main conclusion of the study is that modern networking is characterized by a transition from tightly coupled, vertically integrated hardware-software architectures to horizontal, software-composable, and decentralized models. In such models, network services and functions are dynamically deployed across multiple heterogeneous computing resources under the control of intelligent controllers and orchestrators (centralized tools that automate the management, coordination, deployment, and scaling of distributed containerized applications). Promising areas for future research are proposed, including integration platforms and tools for complex networking, in particular hyperconverged infrastructures; Cloud-native networking plugins for container orchestrators; cloud networking services.

Keywords: hardware networking, software networking, SDN, NFV, P4, SmartNIC, DPU, hyperconverged infrastructure (HCI), IBN, decomposition, cloud native networks, orchestrator.

Введение

Компьютерные сети прошли эволюцию от простых коммуникационных сред в критические, активные и интеллектуальные платформы, которые формируют основу цифровой трансформации практически всех сфер деятельности. Однако исторически сложившаяся архитектура сетей, основанная на специализированных аппаратных устройствах, управляемых проприетарным программным обеспечением (защищенным авторским правом, являющимся частной собственностью разработчика, имеющим закрытый исходный код и строгие ограничения на использование, модификацию и распространение) с распределенной логикой, перестала удовлетворять динамичным требованиям современно-

го общества. Запросы на быстрое масштабирование, высокую степень автоматизации, снижение эксплуатационных расходов и реализация новых сетевых услуг вступили в противоречие с инерционной, сложной в управлении и дорогой традиционной инфраструктурой.

Данная работа направлена на комплексное исследование методов и средств аппаратной и программной организации современных сетей, которые находятся в состоянии активной трансформации. Актуальность темы обусловлена переходом от статичных, аппаратно-центричных моделей к динамичным, программно-управляемым средам, что требует переосмысления как принципов построения, так и инструментария проектировщиков и администраторов сетей.

Объектом исследования являются современные компьютерные сети, рассматриваемые через призму их аппаратно-программной организации.

Предмет исследования – эволюционные и революционные методы, модели, архитектуры и инструментальные средства, определяющие аппаратную реализацию и программное управление сетевыми инфраструктурами.

Цель исследования – системный анализ и классификация современных подходов к аппаратно-программной организации сетей, выявление ключевых трендов и определение перспектив их развития.

Для достижения поставленной цели в работе решаются следующие задачи:

- выполнить анализ эволюции аппаратных платформ сетевых устройств от специализированных решений к открытым и программируемым;
- выполнить анализ трансформации программного стека управления сетями от распределенных CLI-интерфейсов к централизованным контроллерам и декларативным моделям;
- исследовать ключевые интеграционные парадигмы: SDN, NFV, IBN;
- систематизировать инструментальные средства и методологии, такие как IaC и P4, формирующие практики современной сетевой инженерии;
- сформулировать основные тенденции и перспективы развития аппаратно-программной организации сетей.

1 Эволюция аппаратной организации компьютерных сетей: путь к гибкости и открытости

Этап аппаратной организации на базе специализированных интегральных схем (ASIC) и вертикальной интеграции

Более двух десятилетий производительность в ядре сети определялась специализированными интегральными схемами (ASIC). Эти интегральные схемы (СБИС, VLSI), оптимизированные для сверхбыстрой коммутации и маршрутизации пакетов по фиксированным алгоритмам, создавали высокий технологический барьер. Аппаратные платформы ведущих производителей (Cisco, Juniper) представляли собой Black Boxes (закрытые «черные ящики»), где аппаратная часть, системная шина и программная прошивка были тесно интегрированы в единую систему сетевой обработки [1–4].

Этап применения программируемых устройств NPU, FPGA

Для оборудования, требующего большей адаптивности (уровень доступа, агрегации), стали применяться сетевые процессоры (NPU) и программируемые пользователем вентильные матрицы (FPGA) [5, 6]. NPU представляли собой микропроцессоры с набором специализированных инструкций для обработки пакетов, что позволяло пользователю обновлять микрокод. FPGA давали еще большую свободу, позволяя перепрограммировать саму аппаратную логику. Исследованные данные по архитектуре, принципу обработки, производительности, гибкости и стоимости вышеуказанных устройств сетевой обработки сведены в таблицу 1 и таблицу 2.

Таблица 1 – Сравнительные данные по архитектуре и принципу обработки устройств ASIC, NPU, FPGA

Характеристика	ASIC	NPU	FPGA
Архитектура	Жестко зашитая логика. Конвейер предопределен производителем	Многоядерная, многопоточная (PIC-микроядро). SIMD. Гибкий RISC-подобный набор инструкций для пакетов	Матрица программируемых логических вентилей и блоков памяти
Единица обработки	Поток битов через фиксированный конвейер	Микропрограммы (Micro-engines / Threads)	Логическая схема под конкретный алгоритм
Программирование	Отсутствует (ASIC), или прошивка микрокода (микрокод, закрытый от конечного пользователя)	C/C++, микрокод, P4. Загрузка бинарного кода в ядра	HDL (Verilog/VHDL), HLS

Таблица 2 – Производительность, гибкость и стоимость устройств ASIC, NPU, FPGA

Критерий	ASIC	NPU	FPGA
Скорость (Tbps)	Наивысшая (12,8–51,2 Тбит/с). Сквозная задержка фиксирована (<1 мкс)	Высокая (до нескольких Тбит/с). Зависит от сложности кода. Может быть выше, чем FPGA	Средняя/Высокая. Упирается в рабочую частоту схемы (<500 МГц)
Гибкость / Адаптация	Нулевая. Новый стандарт предполагает разработку нового чипа (2–3 года)	Очень высокая. Полная перепрограммируемость логики обработки	Максимальная. Изменение на уровне вентилей. Полная ориентация на конечного пользователя
Энергоэффективность	Абсолютный лидер. Нет накладных расходов на декодирование инструкций	Хорошая, но ниже ASIC. Декодирование инструкций требует энергии	Ниже NPU. Программируемые соединения значительно потребляют энергию
Стоимость единицы оборудования (при соответствующем объеме)	Низкая (миллионы штук). NRE распределен	Высокая (сложная многоядерная структура)	Высокая (малая серия, большой кристалл)

Анализ данных таблицы 1 и таблицы 2 демонстрирует наилучшие показатели производительности, энергоэффективности и стоимости устройств ASIC, однако они по степени адаптивности безнадежно проигрывают сетевым устройствам NPU и FPGA, оставляя за последними перспективы развития.

Программируемый конвейер обработки пакетов: язык P4 и архитектура PISA

Прорывной технологией стал язык программирования плоскостей данных P4 (*Programming Protocol-independent Packet Processors*) и соответствующая архитектура PISA (*Protocol-Independent Switch Architecture*) [7]. В отличие от программирования NPU, P4 описывает что должна делать плоскость данных (обработка заголовков, меток, модификации), а не какой должна быть реализация. Компилятор P4 транслирует программу в конфигурацию для целевой платформы (ASIC, FPGA, программный коммутатор). VLSI Intel Tofino являются первыми массовыми ASIC, полностью программируемыми на P4. Это позволяет:

- определять новые протоколы и форматы заголовков;
- реализовывать сложные, специфичные для приложений политики, например, балансировка нагрузки 5-го уровня (согласно модели OSI, работает балансировка нагрузки 5-го уровня на сеансовом

уровне, обеспечивая управление диалогом, контроль сеансов и восстановление соединения для поддержки устойчивых подключений между клиентом и сервером);

– создавать высоко адаптивные перепрограммируемые ASIC, функциональность которых может меняться программно в течение жизненного цикла устройства.

Полная конвергенция: применение интеллектуальных сетевых карт (SmartNIC) и процессоров данных (DPU)

Крайней формой декомпозиции является перенос сетевых функций непосредственно на серверную платформу. *Smart-сетевые карты (SmartNIC)* и технологически более совершенные *процессоры для обработки данных (Data Processing Unit, DPU)* представляют собой мощные многоядерные системы на кристалле (практически всегда базирующиеся на архитектуре ARM – Advanced RISC Machine – семействе микропроцессорных ядер, основанных на RISC-принципах, отличающихся высокой производительностью) со своей памятью, установленные на PCIe-карте.

Проведенный детальный анализ данных устройств позволил сформулировать базовые характеристики данного семейства, сведенные в таблицу 3.

Таблица 3 – базовые характеристики семейства Smart NIC, DPU

Параметр	SmartNIC (Интеллектуальная сетевая карта)	DPU (Процессор данных)
Основная идея	Сетевая карта, которая берет на себя задачи обработки сетевого трафика, чтобы разгрузить CPU сервера	Программируемый специализированный чип (система на кристалле, SoC), предназначенный для обработки данных в центре обработки данных (ЦОД). Является третьим основным вычислительным элементом (наравне с CPU и GPU)
Основная цель	Ускорение сети и разгрузка CPU. Освобождение процессора от рутинных операций с сетевыми пакетами (например, вычисление контрольных сумм, инкапсуляция туннелей VXLAN)	Инфраструктурная разгрузка (Infrastructure offload). Перенос с CPU всего стека "инфраструктурных" задач (сеть, хранилище, безопасность) в выделенное аппаратное обеспечение для создания изолированной и безопасной "облачной" среды
Архитектура / Состав	1. FPGA (программируемая логика) или ASIC для быстрой обработки пакетов. 2. Несколько ядер ARM (для управления)	Сложная SoC (Система-на-чипе), включающая: 1. Многоядерный CPU (как правило, ARM) для управления. 2. Мощный ASIC для обработки данных. 3. Высокоскоростные интерфейсы памяти и PCIe
Программируемость	Ограниченная. Часто программируется на низком уровне (P4, C/C++ для FPGA) или использует фиксированные функции	Высокая. Поддерживает стандартные языки (C, C++), фреймворки (DOCA от NVIDIA) и запуск полноценных ОС (Linux) непосредственно на DPU
Основные задачи	– разгрузка сетевого стека (TCP/UDP) – обработка туннелей (VXLAN, Geneve) – отбрасывание DDoS-атак на ранних стадиях	Полная виртуализация сети и хранилища – шифрование данных на лету (IPsec, TLS); – изоляция "шумных соседей" в облаке; – управление жизненным циклом сервера (из удаленного центра); – запуск ПО для безопасности (Next-Gen Firewall)
Управление и контроль	Управляется драйверами с хост-сервера (CPU) . Является периферийным устройством	Управляет хост-сервером. Имеет собственную операционную систему и IP-адрес вне зависимости от основного сервера. Может перезагружать, переустанавливать ОС на сервере или полностью его обесточить
Безопасность	Аппаратная изоляция сетевых функций	Аппаратный корень доверия (Root of Trust). Обеспечивает полную изоляцию инфраструктуры (гипервизора, сети хранения) от тенанта (виртуальной машины пользователя), что невозможно на обычном CPU
Область применения	Высоконагруженные веб-серверы, телекоммуникационное оборудование (vRAN), сетевые экраны начального уровня	Крупные публичные облака (AWS, Azure, GCP), частные облака корпоративного уровня (VMware vSphere 8+), HPC (высокопроизводительные вычисления), финансовый сектор (алготрейдинг)

Они выполняют следующие задачи виртуализации и обработки сетевого трафика с аппаратным ускорением:

- выполнение функций виртуального коммутатора (Open vSwitch) и маршрутизатора;
- обеспечение политик безопасности (брандмауэринг, шифрование);
- оркестрация ускоренного доступа к хранилищам (NVMe over Fabrics).

2 Трансформация программного стека управления: централизация, декларативность и автоматизация

Виртуализация сетевых функций (NFV): переход от аппаратных устройств к программным образам

Парадигма NFV декомпозирует не плоскость управления, а сами сетевые функции (firewall, load balancer, WAN accelerator), перенося их с проприетарного аппаратного уровня на стандартные серверы в виде виртуальных машин (VNF) или контейнеров (CNF) [8–10]. Архитектура ETSI NFV определяет ключевые программные компоненты (рисунок 1):

- **менеджер виртуализированных инфраструктур (VIM):** OpenStack, VMware vSphere – управляет физическими ресурсами;
- **оркестратор NFV (NFVO):** Open Source MANO (OSM), платформы вендоров – отвечает за жизненный цикл VNF/CNF (развертывание, масштабирование, обновление).

Таким образом, NFV превращает сеть в набор программных сервисов, которые можно быстро установить, масштабировать и размещать оптимальным образом.

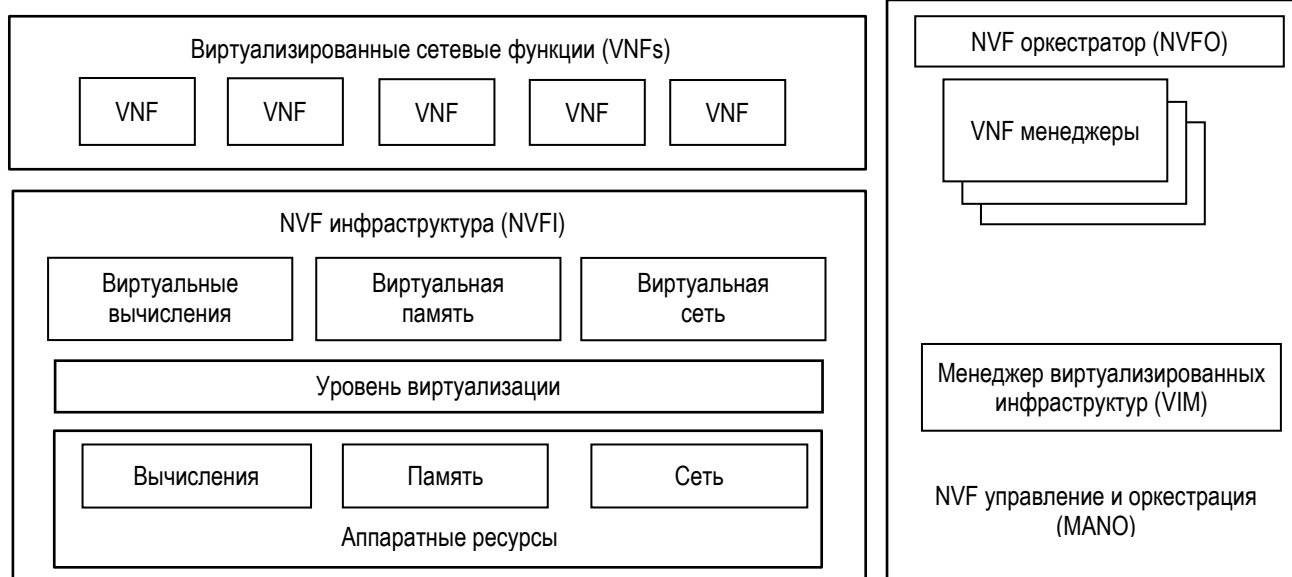


Рисунок 1 – Виртуализация сетевых функций (NFV)

Программно-определяемые сети (SDN)

SDN представляет собой архитектурный подход, отделяющий уровень управления сетью (уровень логики) от уровня передачи данных (аппаратных коммутаторов) [11–13]. Централизованный программный контроллер динамически управляет трафиком, обеспечи-

вая автоматизацию, гибкость, масштабируемость и улучшенную безопасность, заменяя традиционную настройку каждого устройства по отдельности (рисунок 2).

Результаты анализа технологии SDN совместно с технологией NVF приведен в таблице 4.

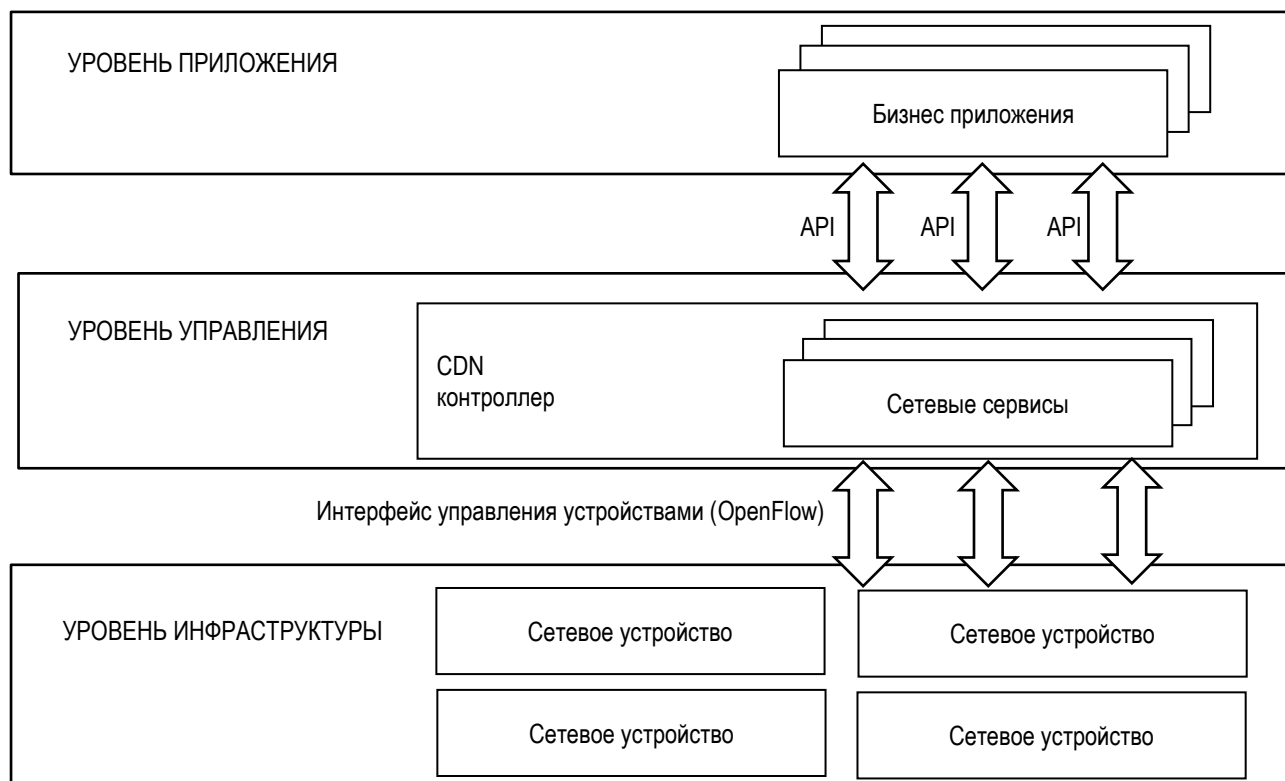


Рисунок 2 – Технология программно-определяемых сетей SDN

Таблица 4 – Концептуальное сравнение технологий SDN и NVF

Критерий сравнения	SDN (Программно-определяемая сеть)	NFV (Виртуализация сетевых функций)
Основная цель	Централизованное управление трафиком, отделение управления от пересылки. Централизация управления. Управление сетью осуществляется через единый контроллер, который видит всю сеть целиком	Замена аппаратных устройств (брандмауэров, маршрутизаторов) программным обеспечением на стандартных серверах. Заставить сеть функционировать дешевле и гибче
Область фокуса (Score)	Контроль и управление потоками трафика (2–3 уровень модели OSI – инфраструктура)	Виртуализация конкретных сервисов (4–7 уровень модели OSI – приложения/функции)
Архитектура	Разделение Control Plane и Data Plane. Централизованный контроллер – <i>разделение плоскостей (Plane Separation)</i> . Уровень управления (Control Plane) отделен от уровня пересылки данных (Data Plane)	Совокупность VNF (Virtual Network Functions), работающих поверх NFVI (инфраструктуры) и управляемых оркестратором MANO
Аппаратная зависимость	Требует поддержки протоколов SDN (например, OpenFlow) на коммутаторах/маршрутизаторах, но может работать с commodity- hardware	Агностична к железу. Работает на любых стандартных промышленных серверах (COTS – Commercial Off-The-Shelf)
Истоки происхождения	Информационные технологии (ИТ), центры обработки данных (ЦОД)	Телекоммуникационная отрасль (операторы связи, такие как Telefonica, AT&T)
Орган стандартизации	Open Networking Foundation (ONF), протокол OpenFlow	European Telecommunications Standards Institute (ETSI)

Сравнительный анализ технологий SDN и NFV по стоимости

Вышеупомянутое семейство технологий в настоящий момент является критическим важным при построении сетевых инфраструктур. Поэтому целесообразным является исследование их параметров с точки зрения их функциональности и эффективности через призму стоимостных характеристик. Ключевые, доступные в открытых источниках [14, 15] приведены в диаграммах на рисунках 3 и 4. Однако прежде, чем перейти

к этому, необходимо зафиксировать ключевое различие, которое пронизывает все источники (таблица 5).

Из результатов, сведенных в таблицу 5, можно сделать следующий вывод: NFV дает прямой и измеримый эффект снижения CAPEX за счет отказа от аппаратного обеспечения; SDN дает более сложно измеримый, но значительный эффект снижения OPEX за счет автоматизации и снижения требований к квалификации персонала.

Таблица 5 – Сравнительные данные природы затрат технологий SDN, NVF

Параметр	SDN (Программно-определяемые сети)	NFV (Виртуализация сетевых функций)
Основная статья экономии	Операционные затраты (OPEX) на управление и администрирование	Капитальные затраты (CAPEX) на специализированное аппаратное обеспечение
Источник затрат	Внедрение контроллеров, обучение персонала, переход на новые модели управления	Вычислительные ресурсы (CPU, RAM) для виртуализации, лицензии ПО
Кто основной бенефициар	Крупные ЦОДы, гиперскейлеры (Google, Facebook), вендорнезависимые компании	Телеком-операторы, сервис-провайдеры, предприятия

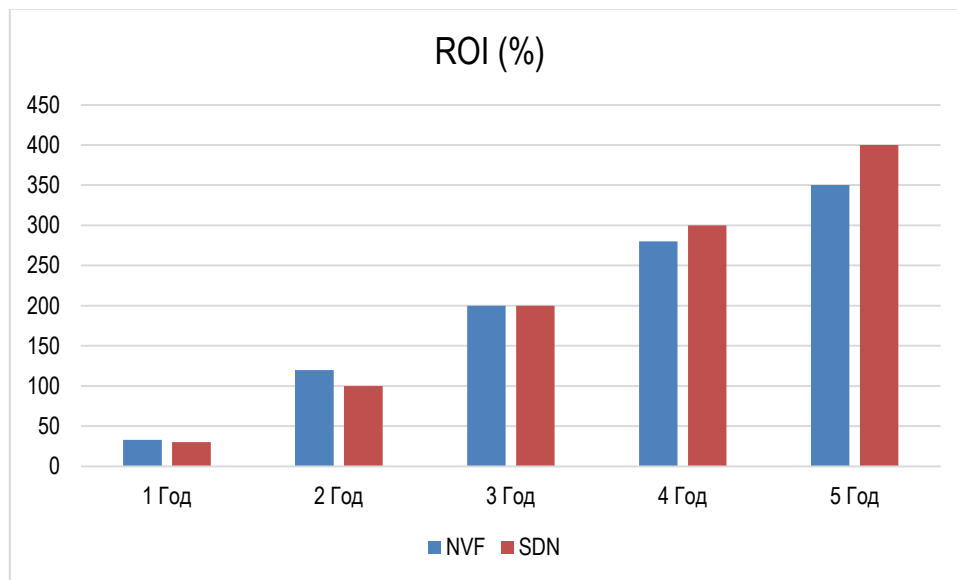


Рисунок 3 – Динамика ROI NFV за 5 лет внедрения и эксплуатации

На рисунке представлена динамика ROI технологии NVF и SGN за пять лет. ROI (Return on Investment) – это показатель, который простыми словами отвечает на вопрос окупаемости вложенных средств и какова полученная прибыль. Таким образом, это мера доходности или убыточности инвестиций, обычно измеряемая в процентах. Чем выше процент ROI, тем выгоднее проект. Исходя из

рисунка 3 очевидно, что обе технологии имеют хорошую прибыльность (приблизительно от 30–33 % до 330–400 %) за пять лет внедрения и эксплуатации.

На рисунке 4 White box экономия – оценочный показатель, прямо не указан в источниках, выведен из контекста. Более дешевые White box существенно повышают ROI.

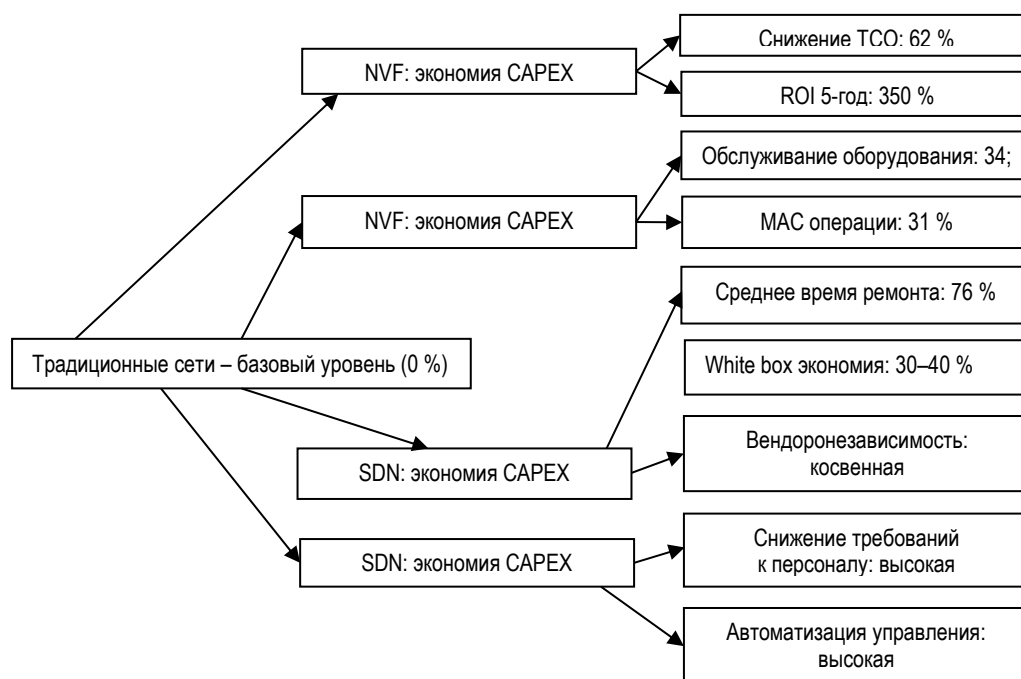


Рисунок 4 – Динамика ROI NFV за пять лет внедрения и эксплуатации

NFV демонстрирует измеримые количественные показатели, SDN – преимущественно качественные.

Сети на основе намерений (Intent-Based Networking – IBN). IBN представляет собой эволюцию SDN, направленную на создание самообучающихся и самоуправляемых систем. Это интеллектуальная сетевая архитектура, использующая искусственный интеллект и автоматизацию для настройки инфраструктуры в соответствии с бизнес-целями. Она переводит административные задачи с ручного управления на декларативный уровень, обеспечивая автоматическое развертывание, мониторинг и оптимизацию [16].

3 Анализ ключевых трендов и перспективы развития

Проведенное исследование позволяет сформулировать и выделить следующие устойчивые и формирующиеся тренды в аппаратно-программной организации сетей:

– *программируемость на всех уровнях* – от языков уровня плоскости данных (P4) до декларативных моделей управления (IaC). Научно-обоснованный анализ подтверждает, что современная программируемость сетей представляет собой вертикально-интегрированный стек. На нижнем уровне язык P4 обеспечивает гибкость обработки трафика, стирая границы между аппаратным и программным обеспечением. Это дает основание и необходимость переходить на принципиально новые подходы к проектированию сетей, масштабируемость и надежность всей инфраструктуры;

– *современная аппаратная основа* сочетает высокопроизводительные программируемые ASIC (PISA) в ядре, DPU на границе серверов и облачные сервисы на периферии. Научно-инженерный анализ определяет выбор платформы экономическими затратами и жесткими требованиями к задержке передачи сетевого трафика;

– *конвергенция сетевой инфраструктуры*. Поведенный научно-обоснованный анализ процессов передачи, хранения и безопасности сетевого трафика говорит о том, что единственным эффективным путем соответствия качества данных процессом потребностями современного общества является слияние, интеграция критически важных ресурсов в единую распределенную вычислительную среду. Таким образом, будущие технологии связаны с платформами, которые могут динамически компоновать необходимые ресурсы из общего пула под нужды конкретного приложения;

– *усиление автономности и интеллекта*. Развитие и интеграция искусственного интеллекта (искусственных нейронных сетей различной архитектуры и иных технологий машинного обучения) создает предпосылки к появлению высокоадаптивных сложных систем, где функции управления, восстановления, оптимизации, восстановления на всех уровнях сетевой модели взаимодействия, аппаратном и программном стеках будут максимально выполняться методами и средствами ИИ. Научный анализ данной проблемы позволил выделить целый пул критически важных задач, которые могли бы эффективно быть решены методами ИИ, реализованными в частности, в нейросетевом базисе (контроль безопасности протоколов передачи, мониторинг сетевого трафика и ресурсов, адаптивное управление нагрузкой, сетевыми устройствами, прогнозирование сетевого трафика и перегрузок и многие другие).

В качестве **перспектив дальнейшего развития в рамках данной работы** можно выделить следующие приоритетные направления:

– *технология «инфраструктура как код (Infrastructure as Code – IaC)* – представляет собой подход к управлению ИТ-инфраструктурой (серверы, компьютерные сети, базы данных) через декларативные конфигурационные файлы, а не ручную настройку; в качестве передовых инструментов можно выделить а) *конфигурационный менеджмент*; б) *оркестрация инфраструктуры* [17];

– *интеграционные платформы и средства комплексной организации сетей*. Яркими представителями их являются: а) *гиперконвергентные инфраструктуры (Hyper-Converged Infrastructure – HCI)* [18, 19]; б) *облачные нативные сетевые плагины для контейнерных оркестраторов* [20, 21]; в) *облачные сетевые сервисы (Cloud Networking Services)* [22].

Заключение

Проведенное исследование современных методов и средств аппаратной и программной организации компьютерных сетей демонстрирует фундаментальный сдвиг от статической, аппаратно-

зависимой инженерии к динамической, программно-управляемой технологии построения и конфигурирования компьютерных сетей и распределенных вычислительных сред. Основные принципы новой парадигмы – декомпозиция (разделение функций), абстракция (скрытие аппаратных деталей) и автоматизация (управление через код и замкнутые циклы).

В такой реализации аппаратная платформа становится стандартизированным, гибким и высокопроизводительным фундаментом, способным выполнять логику обработки данных, заданную программно. Программный стек, в свою очередь, эволюционирует в сторону облачных нативных принципов, централизованной интеллектуальной логики и бесшовной интеграции с системами оркестрации приложений.

Главным результатом такой трансформации является не просто повышение эффективности, а приобретение сетью качества адаптивности. В будущем сеть не будет «проектироваться» в традиционном смысле, а будет программироваться и оркестрироваться как часть единой цифровой среды. Это предъявляет новые требования к специалистам, требуя от них навыков разработки, работы с API, аналитики данных и системного мышления.

Список цитированных источников

- Олифер, В. Г. Компьютерные сети. Принципы, технологии, протоколы : учебник для вузов / В. Г. Олифер, Н. А. Олифер. – СПб. : Питер, 2021. – 944 с.
- Куроуз, Дж. Ф. Компьютерные сети. Нисходящий подход / Дж. Ф. Куроуз, К. В. Росс. – М. : Питер, 2021. – 789 с.
- Таненбаум, Э. Компьютерные сети / Э. Таненбаум, Д. Узеролл. – СПб. : Питер, 2019. – 992 с.
- Петерсон, Л. Л. Компьютерные сети: системный подход / Л. Л. Петерсон, Б. Дэви. – СПб. : БХВ-Петербург, 2018. – 944 с.
- Бруно, Ф. Программирование FPGA для начинающих / Ф. Бруно. – М. : ДМК Пресс, 2022. – 304 с.
- Dhar, S. FPGA Accelerated FPGA Placement / S. Dhar, L. Singhal, M. Iyer // 29th International Conference on Field Programmable Logic and Applications (FPL). – Barcelona, Spain, 2019. – P. 404–410.
- P4: Programming Protocol-Independent Packet Processors / P. Bosshart, D. Daly, G. Bibb [et al.] // ACM SIGCOMM Computer Communication Review. – 2014. – Vol. 44, № 3. – P. 87–95.
- A Comprehensive Survey of Network Function Virtualization / B. Yi, X. Wang, K. Li, M. Huang // Computer Networks. – 2018. – Vol. 133. – P. 212–262.
- Mijumbi, R. Network Function Virtualization: State-of-the-art and Research Challenges / R. Mijumbi, J. Serrat, J. L. Gorricho // IEEE Communications Surveys and Tutorials. – 2015. – Vol. 18. – P. 236–262.
- Callegati, F. Virtual Networking Performance in OpenStack Platform for Network Function Virtualization / F. Callegati, W. Cerroni, C. Contoli // Journal of Electrical and Computer Engineering. – 2016. – P. 1–15.
- Найдхарт, Т. Программируемые сети: от идеи к реализации. Основы автоматизации сетей с помощью Linux, Python, Ansible и OpenDaylight / Т. Найдхарт. – М. : ДМК Пресс, 2019. – 264 с.
- Kim, H. Improving Network Management with Software Defined Networking / H. Kim, N. Feamster. // IEEE Communications Magazine. – 2013. – Vol. 51, № 2. – P. 114–119.
- Kreutz, D. Software-Defined Networking: A Comprehensive Survey / D. Kreutz, F. M. Ramos, P. E. Verissimo // Proceedings of the IEEE. – 2015. – Vol. 103, № 1. – P. 14–76.
- SD-WAN ROI: How to Quantify Cost Savings and Operational Efficiency. – URL: <https://www.scc.com/insights/it-services/networking/sd-wan-roi-how-to-quantify-cost-savings-and-operational-efficiency> (date of access: 27.12.2025).
- NFV Will Drive Enterprise SDN Adoption. – URL: <https://www.sdxcentral.com/analysis/nfv-will-drive-enterprise-sdn-adoption> (date of access: 27.12.2025).
- Prakash, V. Intent-Based Networking for Modern Infrastructure / V. Prakash, R. Singh. – USA, Sebastopol : O'Reilly Media, 2023. – 310 p.
- Моррис, К. Инфраструктура как код: управление серверами в облаке / К. Моррис ; пер. с англ. А. С. Киселева. – СПб. : Питер, 2021. – 384 с.

18. Halabi, S. Hyperconverged Infrastructure Data Centers: Demystifying HCI / S. Halabi. – USA, Indianapolis : Cisco Press, 2019. – 544 p.
19. Blokdyk, G. Hyperconverged Infrastructure: A Complete Guide / G. Blokdyk. – Toronto : 5STARCOOKS, 2021. – 312 p.
20. Project Calico. Calico documentation. – URL: <https://projectcalico.docs.tigera.io> (date of access: 15.12.2025).
21. Cilium – Cloud Native, eBPF-based Networking, Observability, and Security. – URL: <https://cilium.io> (date of access: 22.12.2025).
22. AWS. What is Cloud Networking? – URL: <https://aws.amazon.com/what-is/cloud-networking> (date of access: 28.12.2025).
10. Callegati, F. Virtual Networking Performance in OpenStack Platform for Network Function Virtualization / F. Callegati, W. Cerroni, C. Contoli // Journal of Electrical and Computer Engineering. – 2016. – P. 1–15.
11. Najdhart, T. Programmiruemye seti: ot idei k realizacii. Osnovy avtomatizacii setej s pomoshch'yu Linux, Python, Ansible i OpenDaylight / T. Najdhart. – M. : DMK Press, 2019. – 264 s.
12. Kim, H. Improving Network Management with Software Defined Networking / H. Kim, N. Feamster. // IEEE Communications Magazine. – 2013. – Vol. 51, № 2. – P. 114–119.
13. Kreutz, D. Software-Defined Networking: A Comprehensive Survey / D. Kreutz, F. M. Ramos, P. E. Verissimo // Proceedings of the IEEE. – 2015. – Vol. 103, № 1. – P. 14–76.
14. SD-WAN ROI: How to Quantify Cost Savings and Operational Efficiency. – URL: <https://www.scc.com/insights/it-services/networking/sd-wan-roi-how-to-quantify-cost-savings-and-operational-efficiency> (date of access: 27.12.2025).
15. NFV Will Drive Enterprise SDN Adoption. – URL: <https://www.sdxcentral.com/analysis/nfv-will-drive-enterprise-sdn-adoption> (date of access: 27.12.2025).
16. Prakash, V. Intent-Based Networking for Modern Infrastructure / V. Prakash, R. Singh. – USA, Sebastopol : O'Reilly Media, 2023. – 310 p.
17. Morris, K. Infrastruktura kak kod: upravlenie serverami v oblake / K. Morris ; per. s angl. A. S. Kiseleva. – SPb. : Piter, 2021. – 384 s.
18. Halabi, S. Hyperconverged Infrastructure Data Centers: Demystifying HCI / S. Halabi. – USA, Indianapolis : Cisco Press, 2019. – 544 p.
19. Blokdyk, G. Hyperconverged Infrastructure: A Complete Guide / G. Blokdyk. – Toronto : 5STARCOOKS, 2021. – 312 p.
20. Project Calico. Calico documentation. – URL: <https://projectcalico.docs.tigera.io> (date of access: 15.12.2025).
21. Cilium – Cloud Native, eBPF-based Networking, Observability, and Security. – URL: <https://cilium.io> (date of access: 22.12.2025).
9. Mijumbi, R. Network Function Virtualization: State-of-the-art and Research Challenges / R. Mijumbi, J. Serrat, J. L. Gorricho // IEEE Communications Surveys and Tutorials. – 2015. – Vol. 18. – P. 236–262.

References

Материал поступил 06.02.2026, одобрен 16.02.2026, принят к публикации 17.02.2026