Санкт-Петербургский
государственный
университет
www.spbu.ru

ВИРТУАЛЬНАЯ ЛАБОРАТОРИЯ ДЛЯ ОБЕСПЕЧЕНИЯ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ ВЫЧИСЛЕНИЙ

Профессор кафедры компьютерного моделирования и многопроцессорных систем

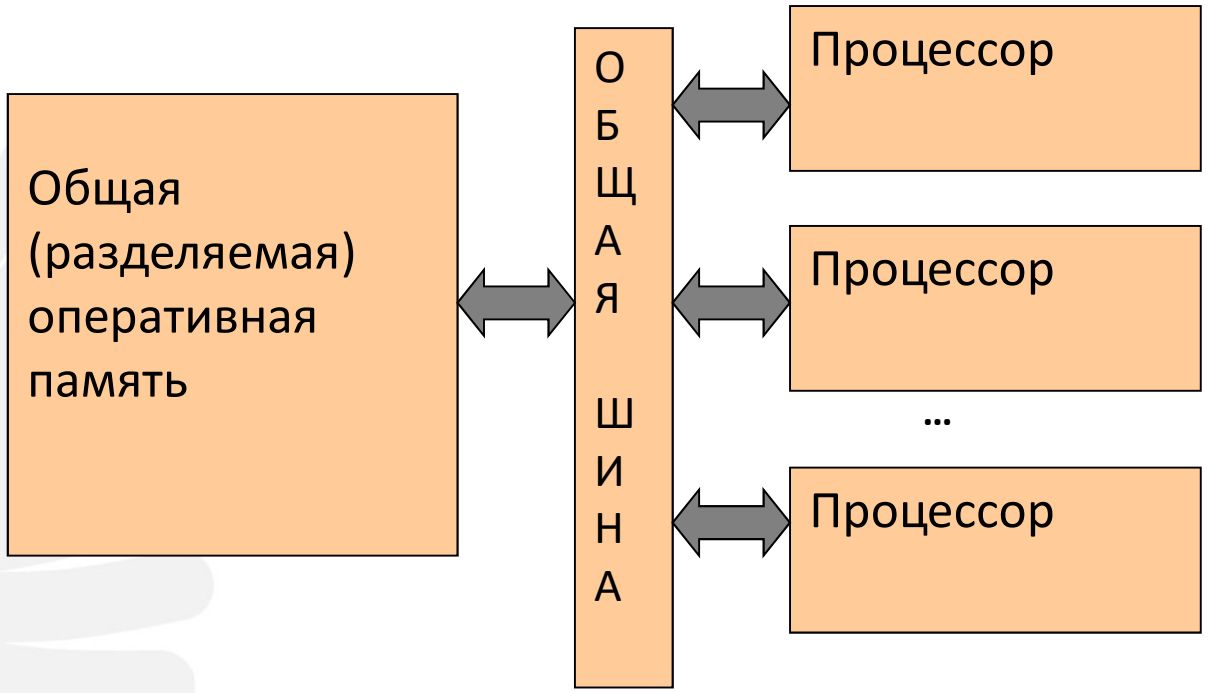
А.Б.Дегтярев

БАЗОВЫЕ ТИПЫ АРХИТЕКТУР

- SMP
- MPP
- NUMA
- Hybrid

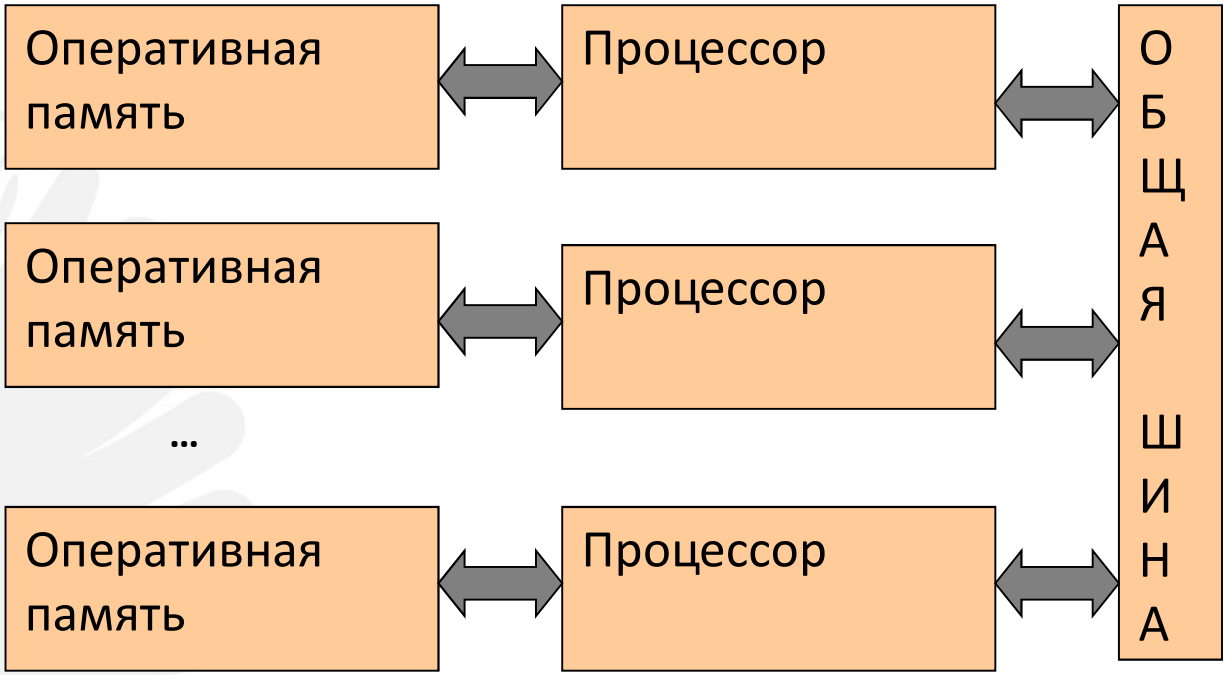
БАЗОВЫЕ ТИПЫ АРХИТЕКТУР

- SMP
- MPP
- NUMA
- Hybrid



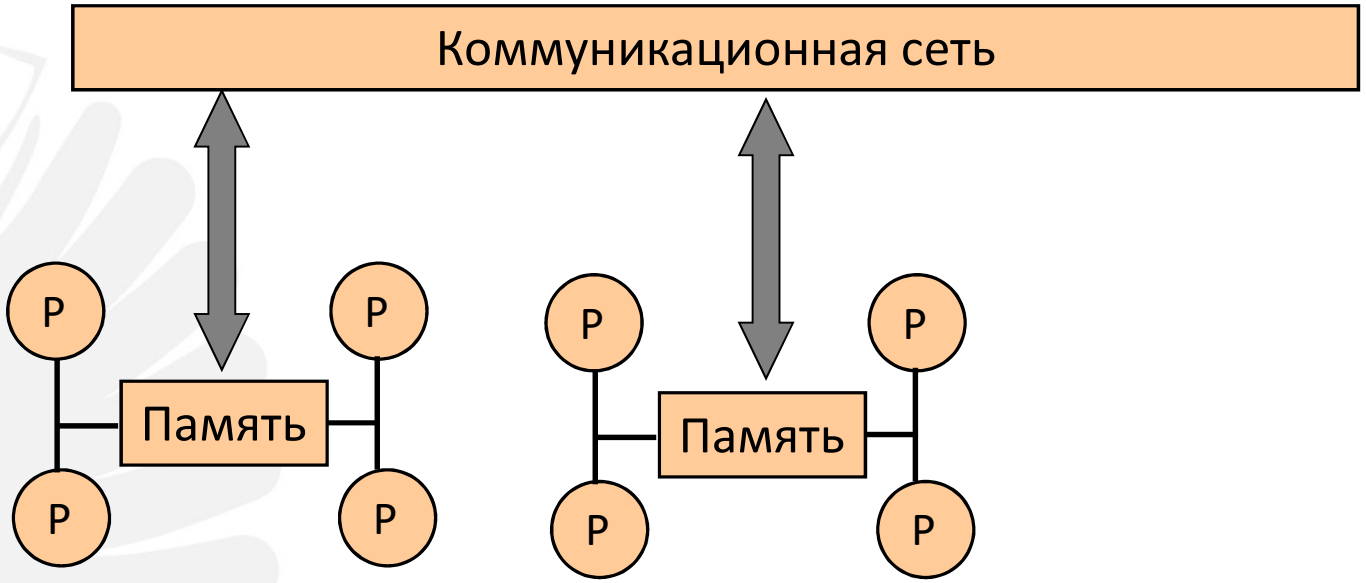
БАЗОВЫЕ ТИПЫ АРХИТЕКТУР

- SMP
- MPP
- NUMA
- Hybrid



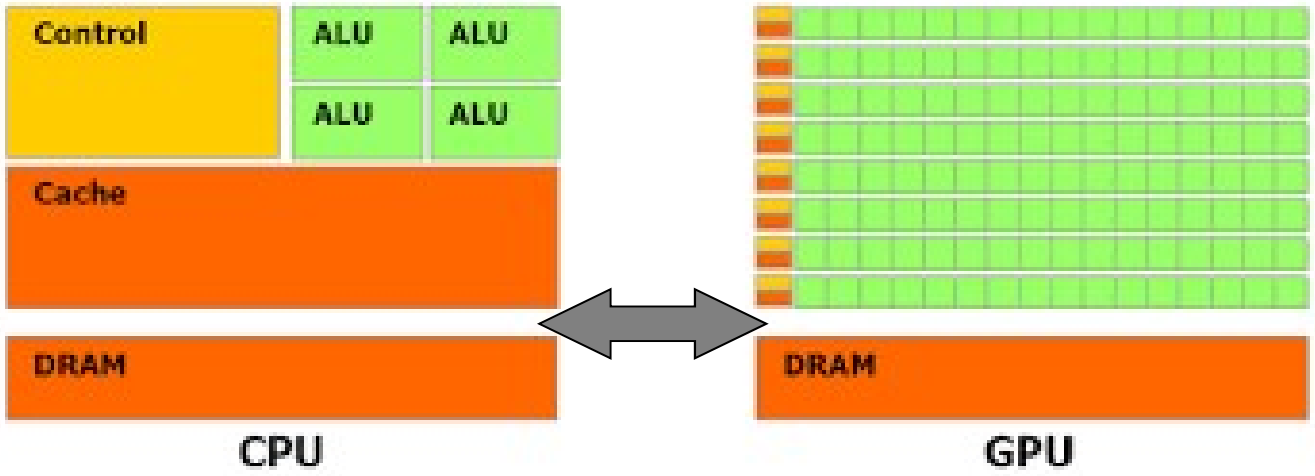
БАЗОВЫЕ ТИПЫ АРХИТЕКТУР

- SMP
- MPP
- NUMA
- Hybrid

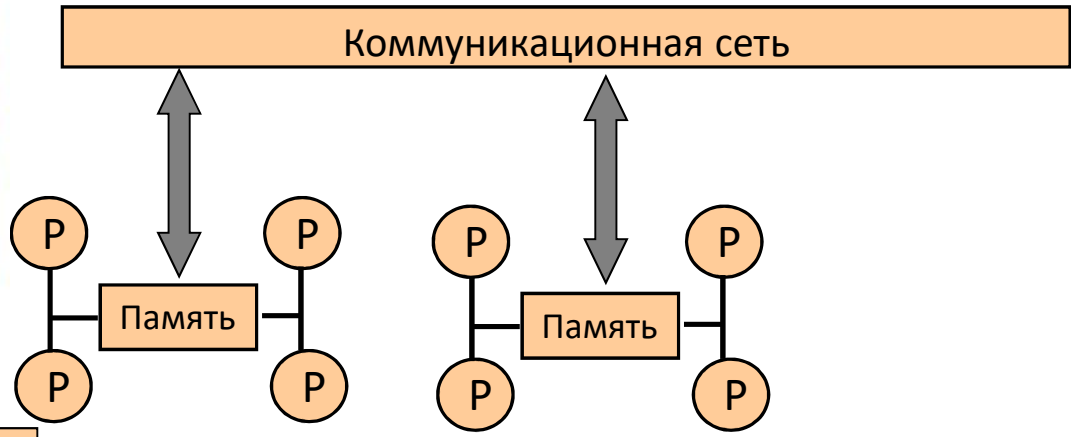


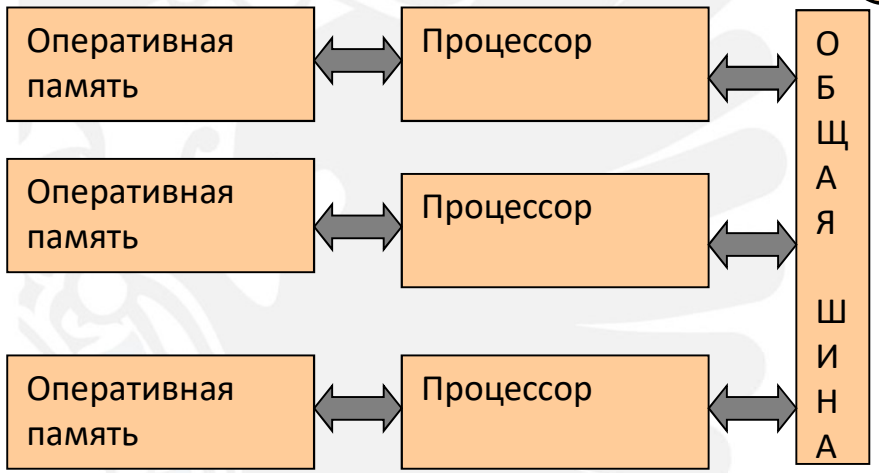
БАЗОВЫЕ ТИПЫ АРХИТЕКТУР

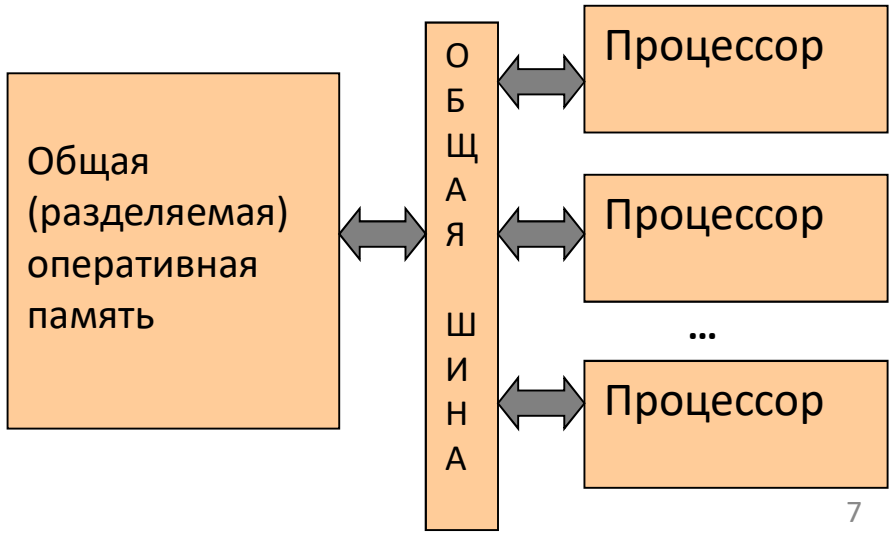
- SMP
- MPP
- NUMA
- Hybrid




Control
ALU
ALU
ALU
ALU
Cache
DRAM
CPU
DRAM
GPU


Коммуникационная сеть
P
P
Память
P
P
P
P
Память
P
P


Оперативная память
Процессор
Оперативная память
Процессор
Оперативная память
Процессор
ОБЩАЯ ШИНА


Общая (разделяемая) оперативная память
ОБЩАЯ ШИНА
Процессор
Процессор
...
Процессор

ФОРМАЛЬНАЯ МОДЕЛЬ УСКОРЕНИЯ

$$S_p = \frac{T_1}{(\alpha_1 + \alpha_2/k + \alpha_3/p)T_1 + t_d}$$

T_1 – время выполнения алгоритма на одном процессоре
α_1 – доля последовательных вычислений
α_2 – доля вычислений со средней степенью параллелизма k<p
α_3 – доля вычислений с максимальной степенью параллелизма p
t_d – общее время на подготовку данных

$$\alpha_1 + \alpha_2 + \alpha_3 = 1$$

УСКОРЕНИЕ С УЧЕТОМ НАКЛАДНЫХ РАСХОДОВ

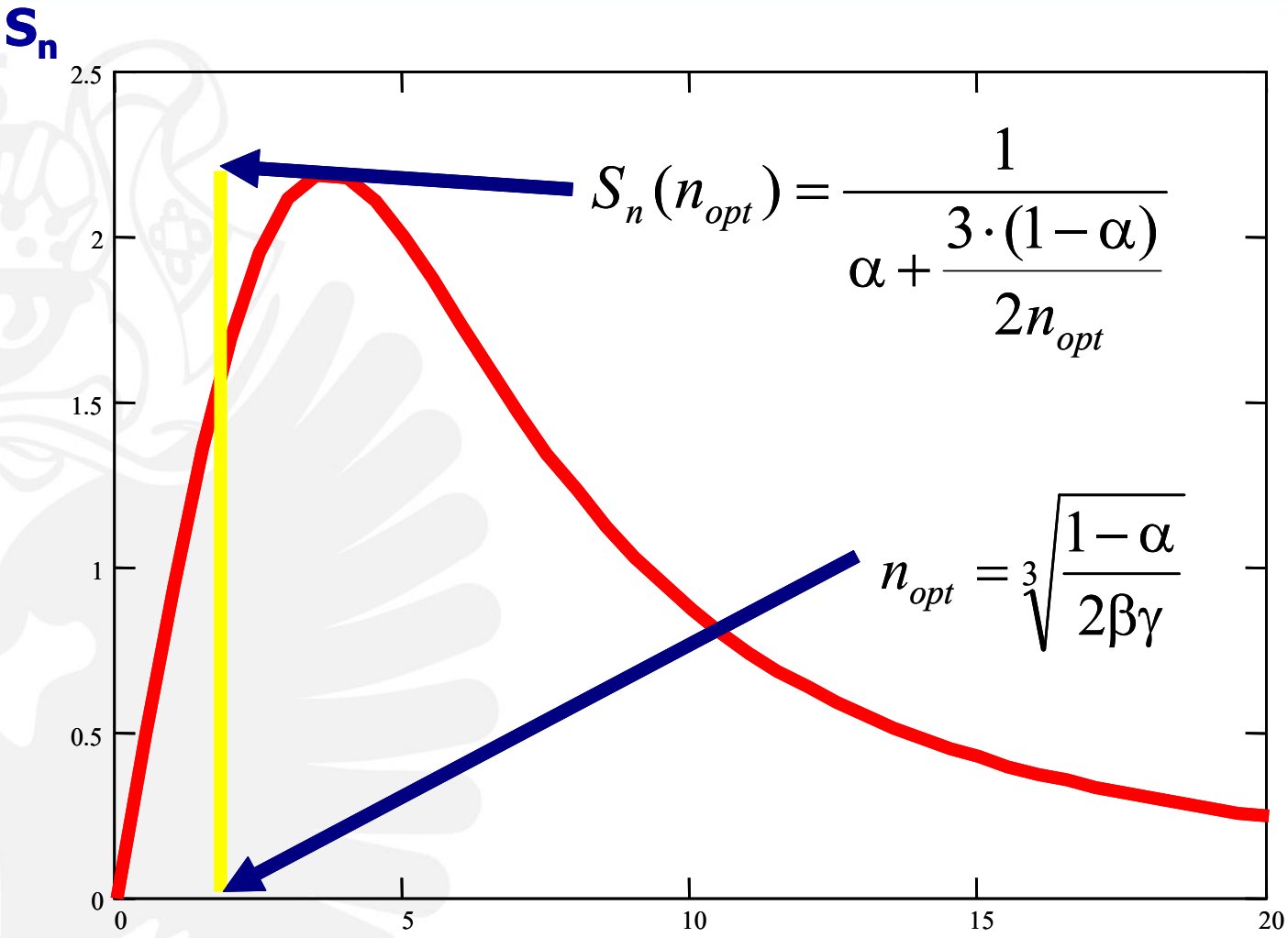
$$S_p = \frac{p}{1 - \alpha + \alpha p + \beta\gamma p^3}$$

α – доля последовательных вычислений

β – коэффициент, характеризующий диаметр системы

γ – коэффициент, характеризующий отношение мощности вычислительного узла к производительности межпроцессорного соединения

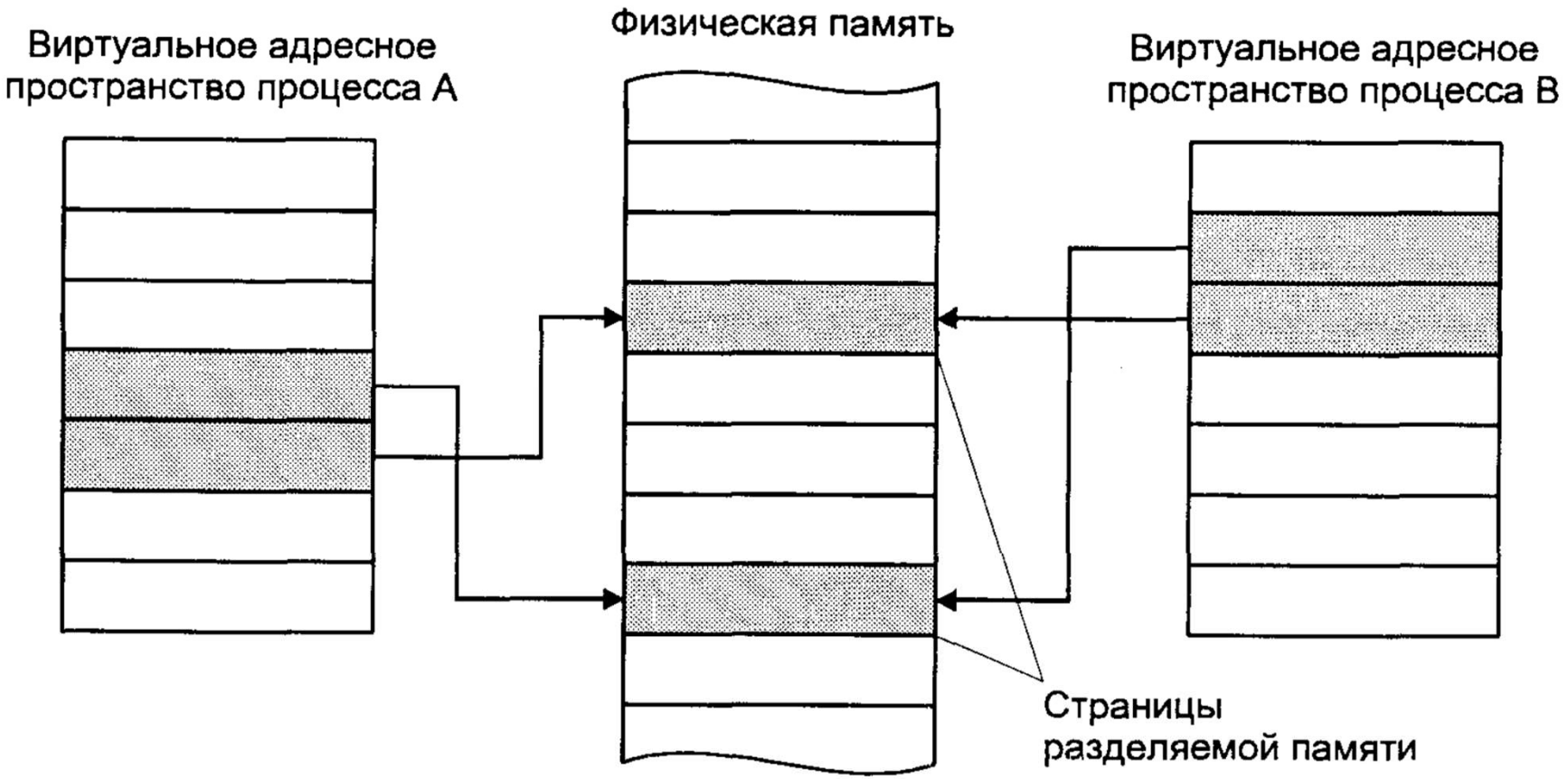
УСКОРЕНИЕ С УЧЕТОМ НАКЛАДНЫХ РАСХОДОВ



РАЗДЕЛЯЕМАЯ ПАМЯТЬ

```
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
int shmget (key_t key, int size,
  int shmflag);
char *shmat(int shmid, char
  *shmaddr, int shmflag);
int shmdt(char *shmaddr);
```

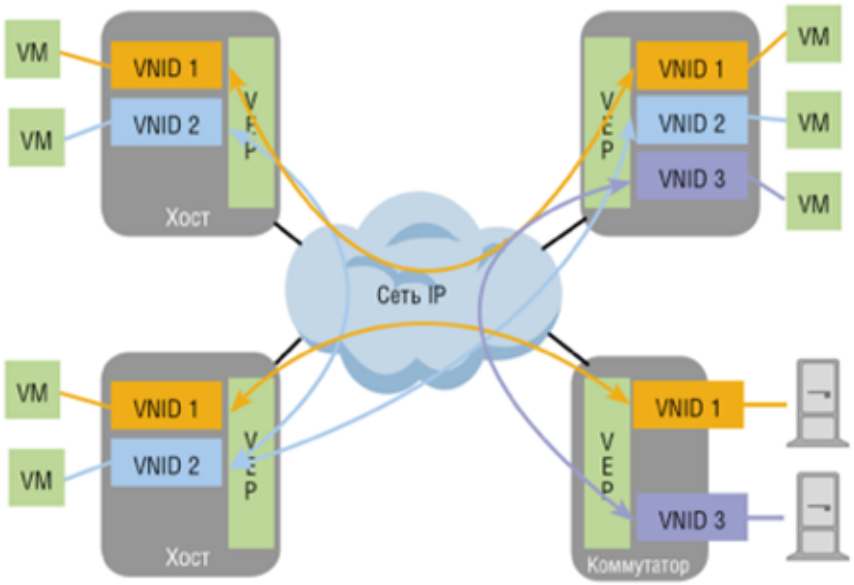
СОВМЕСТНОЕ ИСПОЛЬЗОВАНИЕ РАЗДЕЛЯЕМОЙ ПАМЯТИ

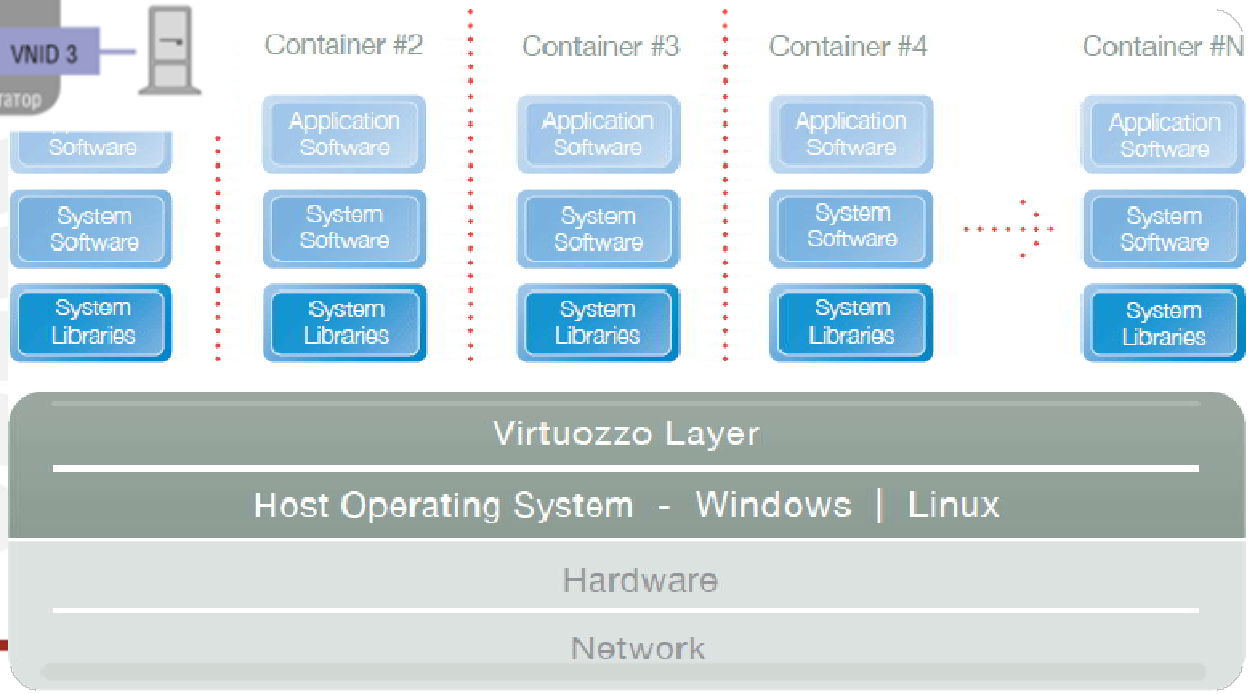


ВИРТУАЛИЗАЦИЯ

Сокрытие настоящей реализации какого-либо процесса или объекта от истинного его представления для того, кто им пользуется.

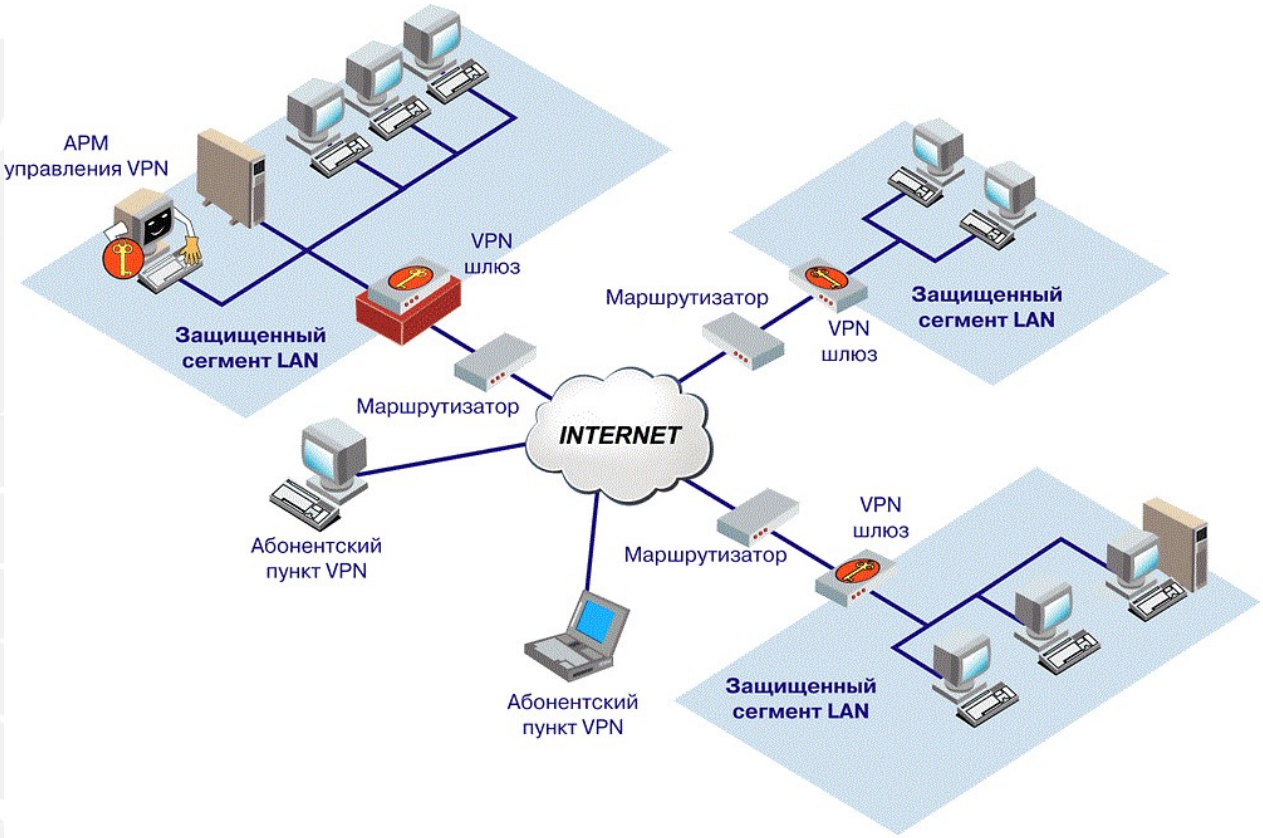
ВИРТУАЛЬНЫЕ МАШИНЫ ОБЩЕГО НАЗНАЧЕНИЯ





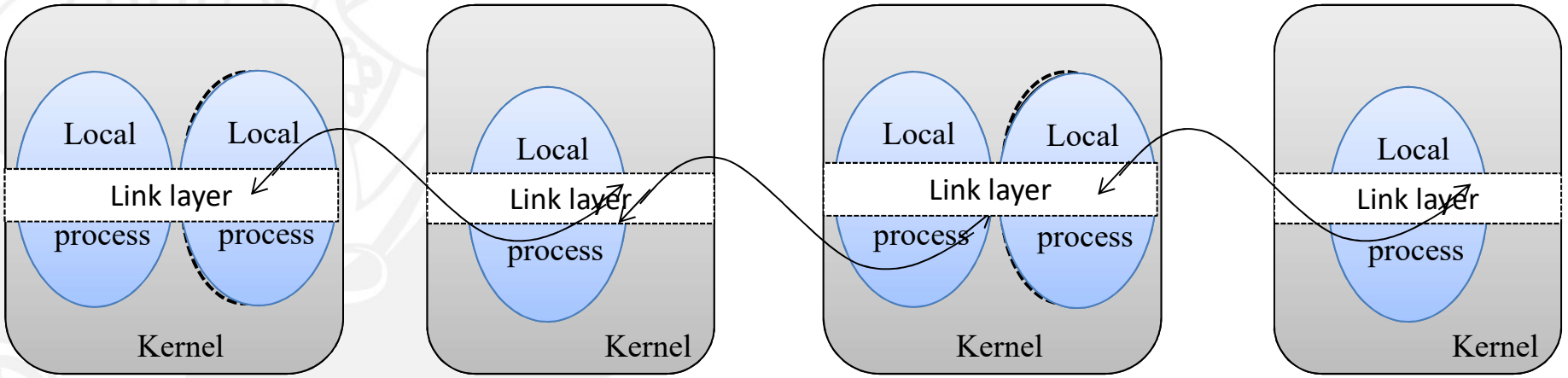
ВИРТУАЛИЗАЦИЯ СЕТЕЙ

Виртуализация сетей предоставляет различным группам пользователей доступ к виртуальным ресурсам на удаленной системе через виртуальную сеть..



Корпоративная VPN

МИГРАЦИЯ ПРОЦЕССОВ



Remote - Пользовательская часть

Deputy - Системная часть

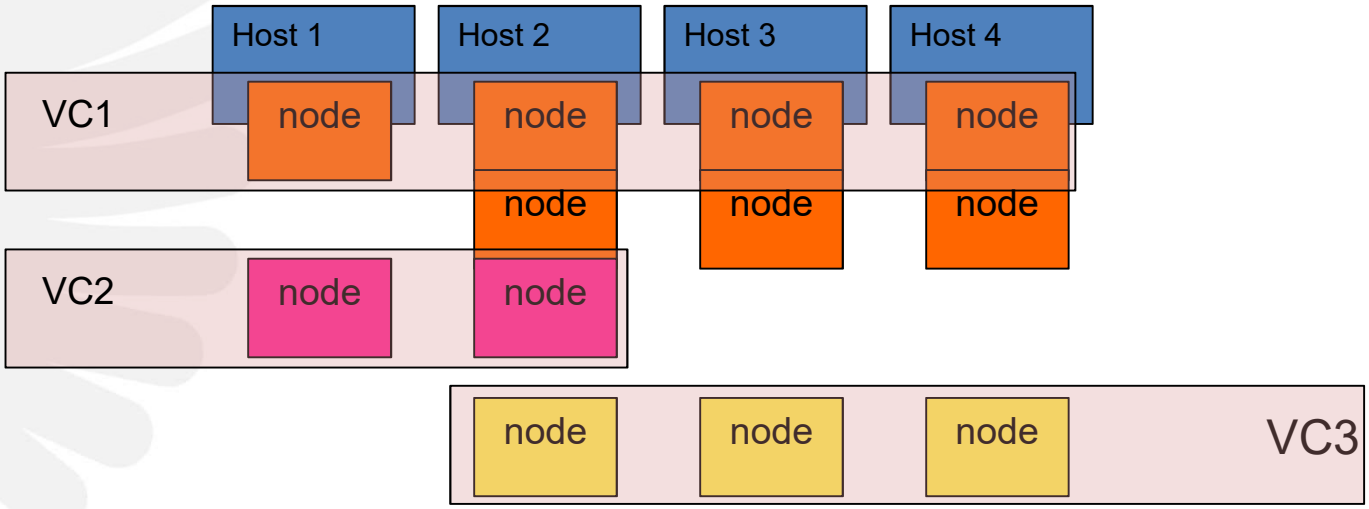
ВИРТУАЛЬНЫЙ СУПЕРКОМПЬЮТЕР

Создание специально сконфигурированных виртуальных систем

- Конфигурация инфраструктуры под требования приложения, а не наоборот

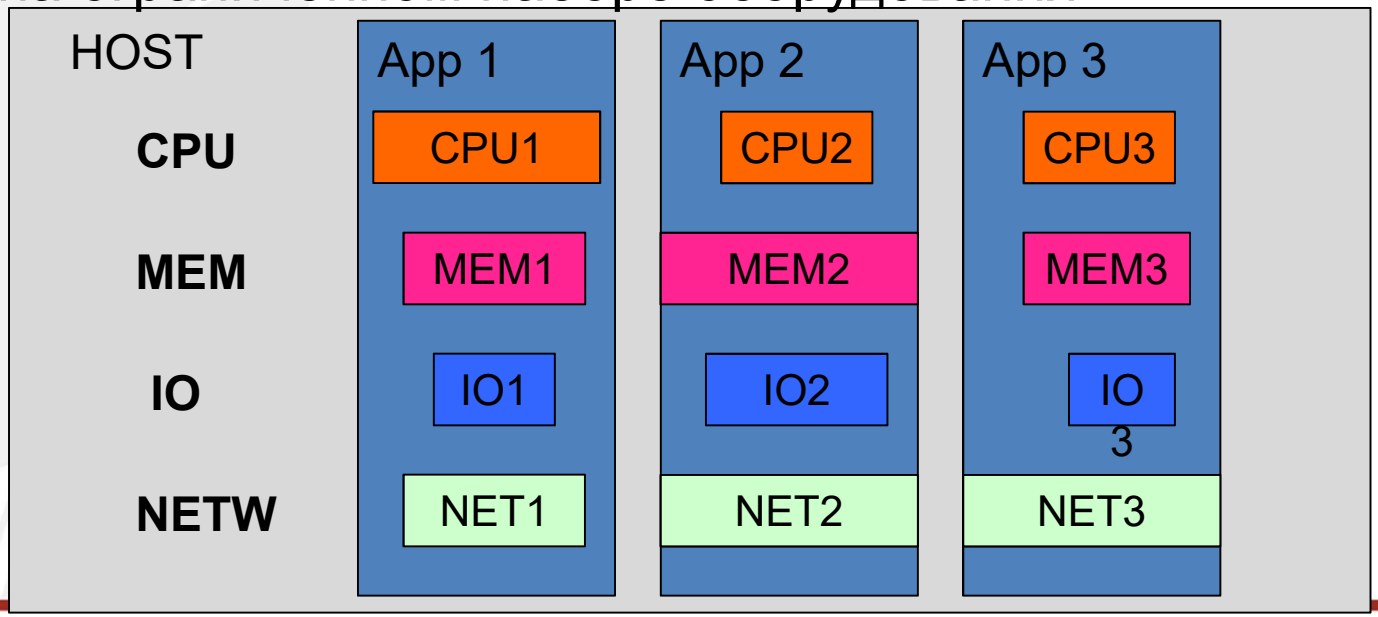
Виртуализация ресурсов

- Виртуальные кластеры
- Легковесная виртуализация с малыми накладными расходами
- Гибкая настройка инфраструктуры



ПОЧЕМУ ВИРТУАЛЬНЫЙ КЛАСТЕР?

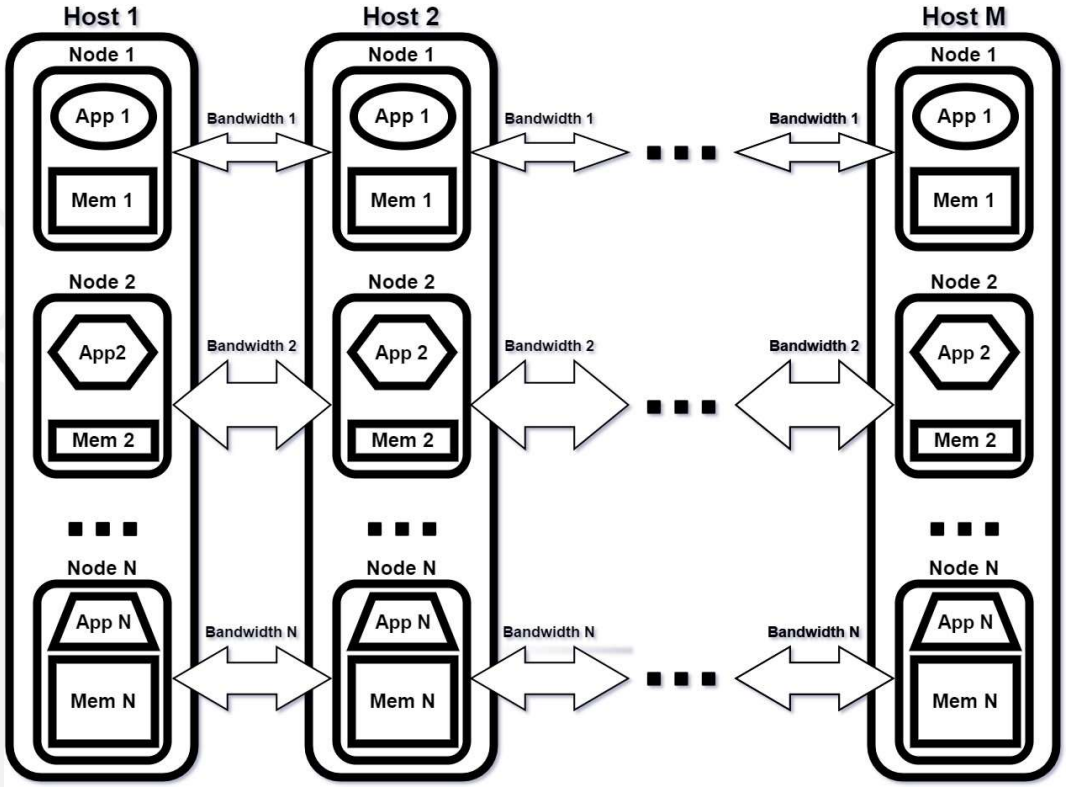
- Точный контроль над распределенными ресурсами (CPU, память и т.д.)
- Приложения получают именно то, что им нужно (или то, что запрашивают): одному требуется быстрый дисковый в/в и не так много CPU, другому – быстрая сеть и быстрый CPU без дискового в/в и т.д.
- Емкость невостребованных ресурсов, доступных для других приложений на ограниченном наборе оборудования



ПРОБЛЕМЫ СОЗДАНИЯ ВИРТУАЛЬНОГО КЛАСТЕРА

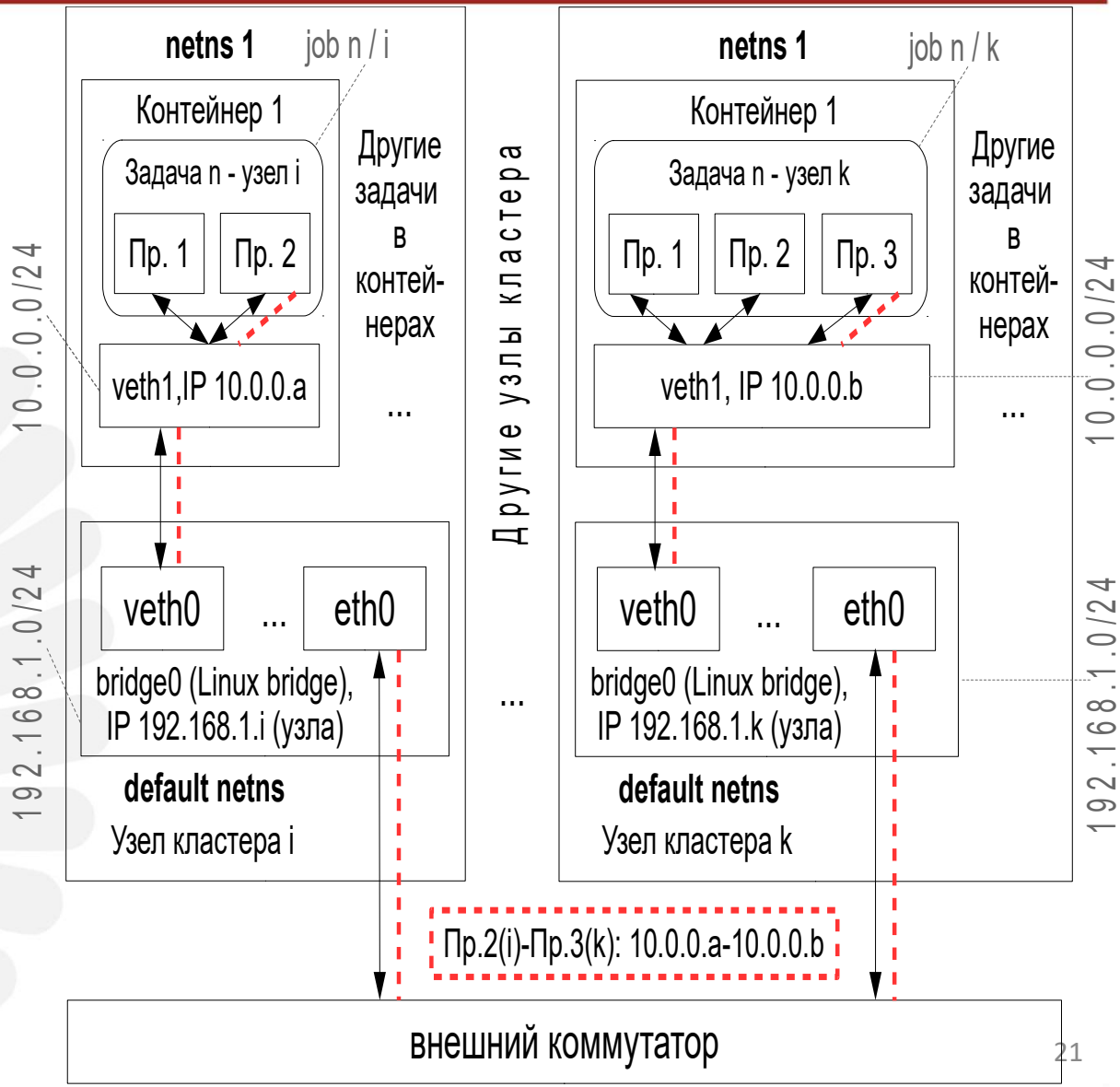
- Обеспечение динамической миграции процессов и контейнеров: C/R, сохранение состояния процессов и пр.
- Обеспечение оптимальной кластеризации узлов виртуального кластера
- Обеспечение производительности виртуальной сети: выбор архитектуры, интерфейсов и пр.
- Обеспечение легкости организации и безопасного доступа

Виртуальный суперкомпьютер: использование ресурсов

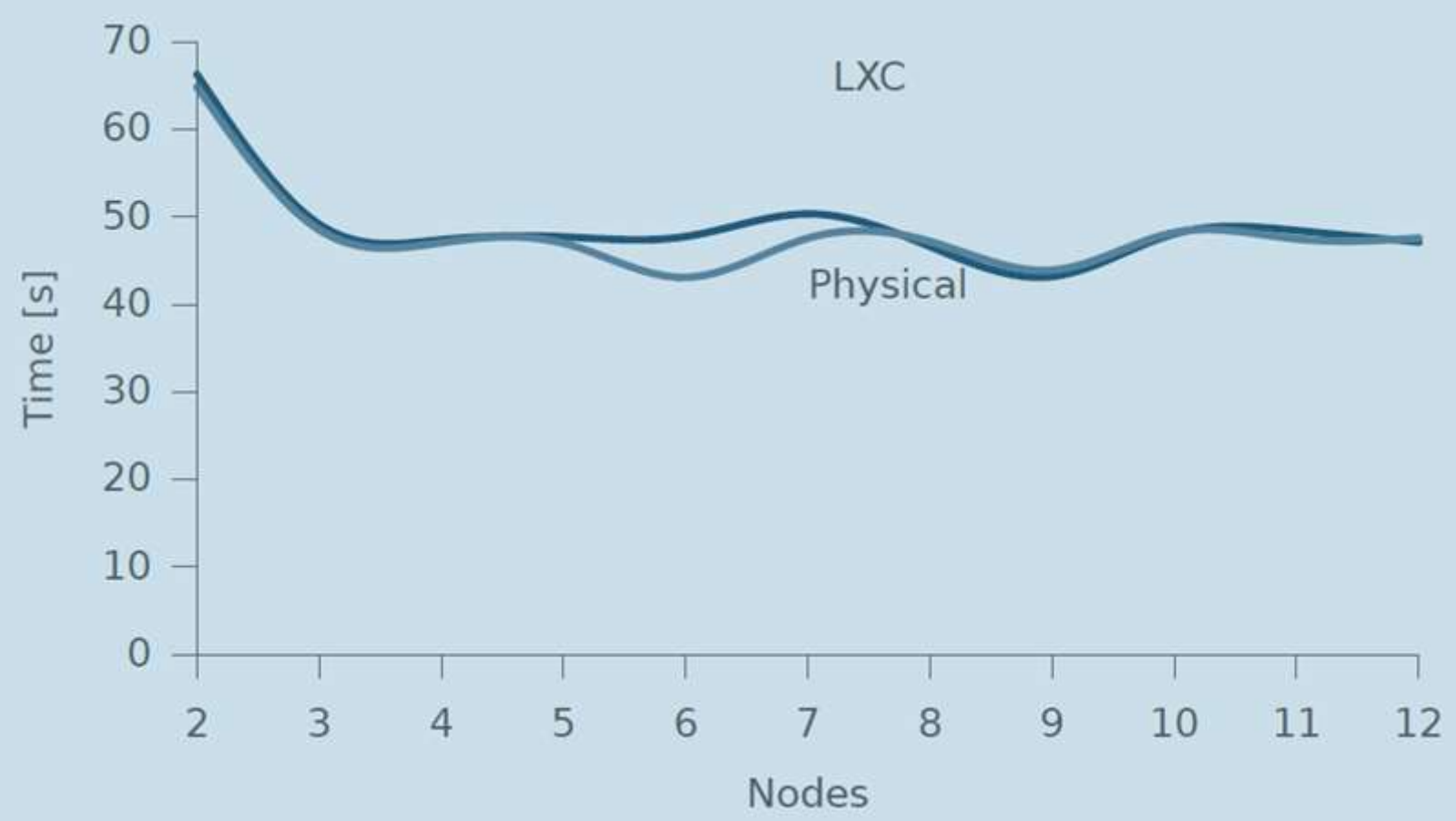


АРХИТЕКТУРА СИСТЕМЫ ДЛЯ МИГРАЦИИ ЗАДАЧ

- Выделение набора свободных IP-адресов
- Развертывание ВК
- Определение узлов для задачи (IP)
- Запуск скрипта задачи в рамках контейнера
- При необходимости checkpoint
- В нужный момент restart с предварительным восстановлением настройки контейнера
- Checkpoint/Restart с помощью CRIU (Checkpoint/Restore In Userspace)

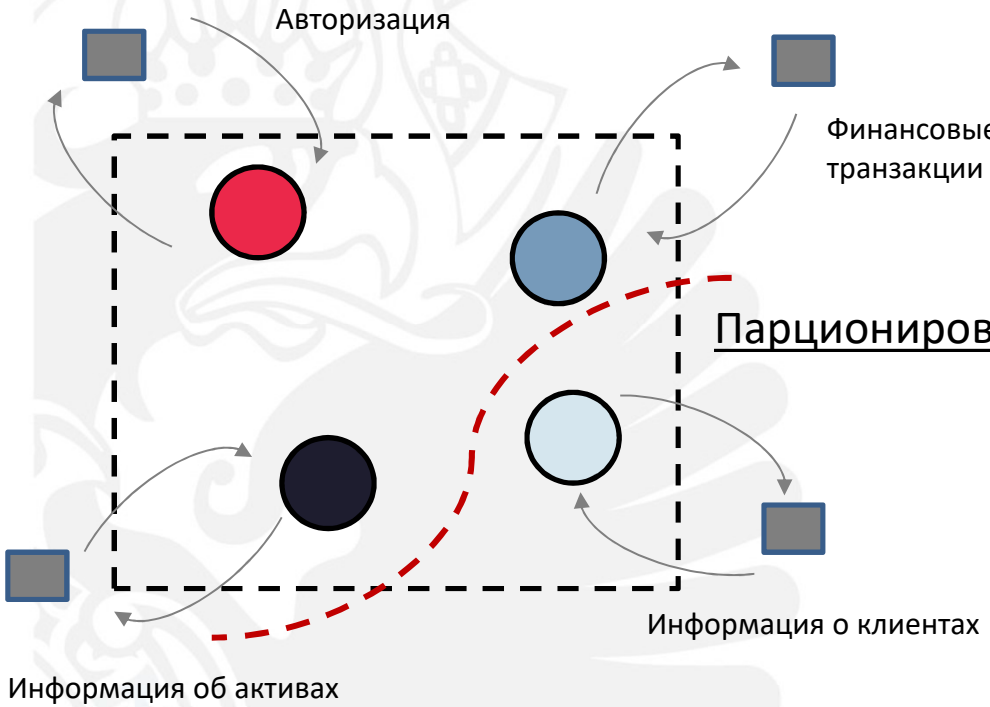


OpenFOAM RUN TIME

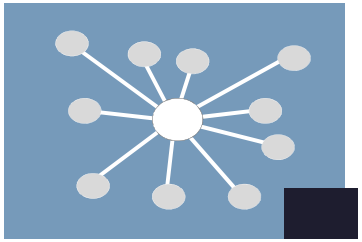


ДЕЦЕНТРАЛИЗАЦИЯ И РАСПРЕДЕЛЕННОСТЬ

Взаимодействующие между собой участники объединяются в сеть, которая может иметь разную топологию – централизованную, децентрализованную, распределенную. Степень распределенности и децентрализации может различаться

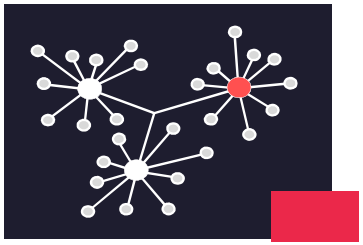


Сети могут распадаться на фрагменты (парционирование), узлы могут оказываться недоступными из-за своих SLA – все эти причины влияют на совокупность данных в сети, их целостность и доступность.

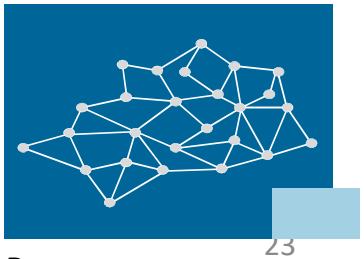


Централизованная сеть

В зависимости от особенностей сети ее характеристики влияют на данные:



Децентрализованная сеть



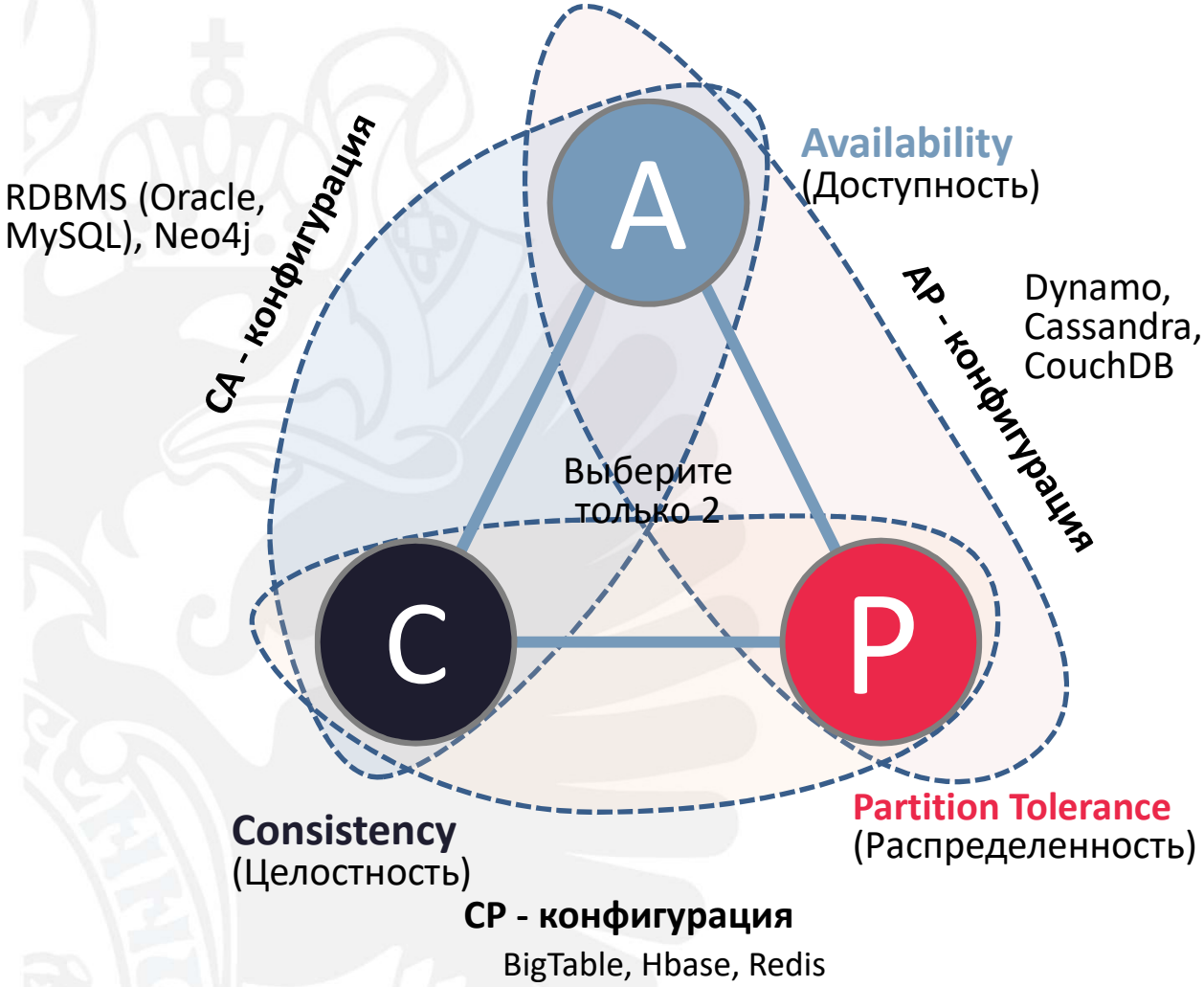
Распределенная сеть

Механизм консенсуса (согласования) данных зависит от уровня децентрализации и распределенности и влияет на TPS и задержку

SLA (Service Level Agreement) – уровень мощности и производительности отдельных узлов

TPS – transaction per seconds, производительность вычислительной системы по отношению к числу обрабатываемых за одну секунду транзакций

CAP-ТЕОРЕМА/PACELC ТЕОРЕМА



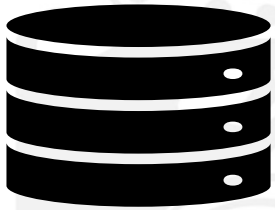
Теорема CAP (теорема Брюера) — эвристическое утверждение о том, что в любой реализации распределённых вычислений возможно обеспечить не более двух из трёх следующих свойств: согласованность данных (во всех вычислительных узлах в один момент времени данные не противоречат друг другу), доступность (любой запрос к распределённой системе завершается корректным откликом), устойчивость к разделению (partition tolerance)

BASE-архитектура Basically Available, Soft-state, Eventually consistent — базовая доступность, неустойчивое состояние, согласованность в конечном счёте

ОРГАНИЗАЦИЯ ХРАНЕНИЯ ДАННЫХ

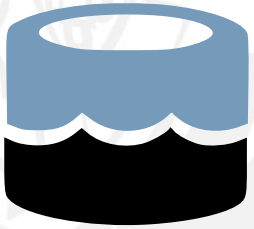
Организация системы хранения данных, синхронизация данных, порядок вставки данных в общий для узлов реестр накладывает ограничения на архитектуру данных и общий подход к интеграции

Distributed Ledger Technologies (технологии распределенных реестров), включают в себя класс блокчейн решений



Реляционные базы данных

Распределенность источников данных решается за счет репликации и ETL загрузок

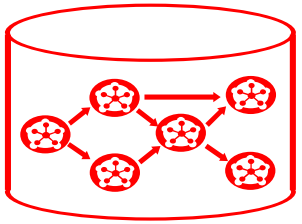


Озера данных (NoSQL базы)

Отказ от полных принципов ACID, переход к BASE архитектуре

Блокчейн

Хранение данных в блоках, на которые накладываются ограничения. Фактически хранение блоков представляет собой линейный связанный список



DAG – базированные решения

Хранение данных в DAG – графовых базах, в которых кроме самих данных хранятся связи, допускаются ветвления

Механизм консенсуса (согласования) данных во многом определяется порядком чтения и записи данных сообщений (транзакций), различные консенсусы используют разные алгоритмы записи данных

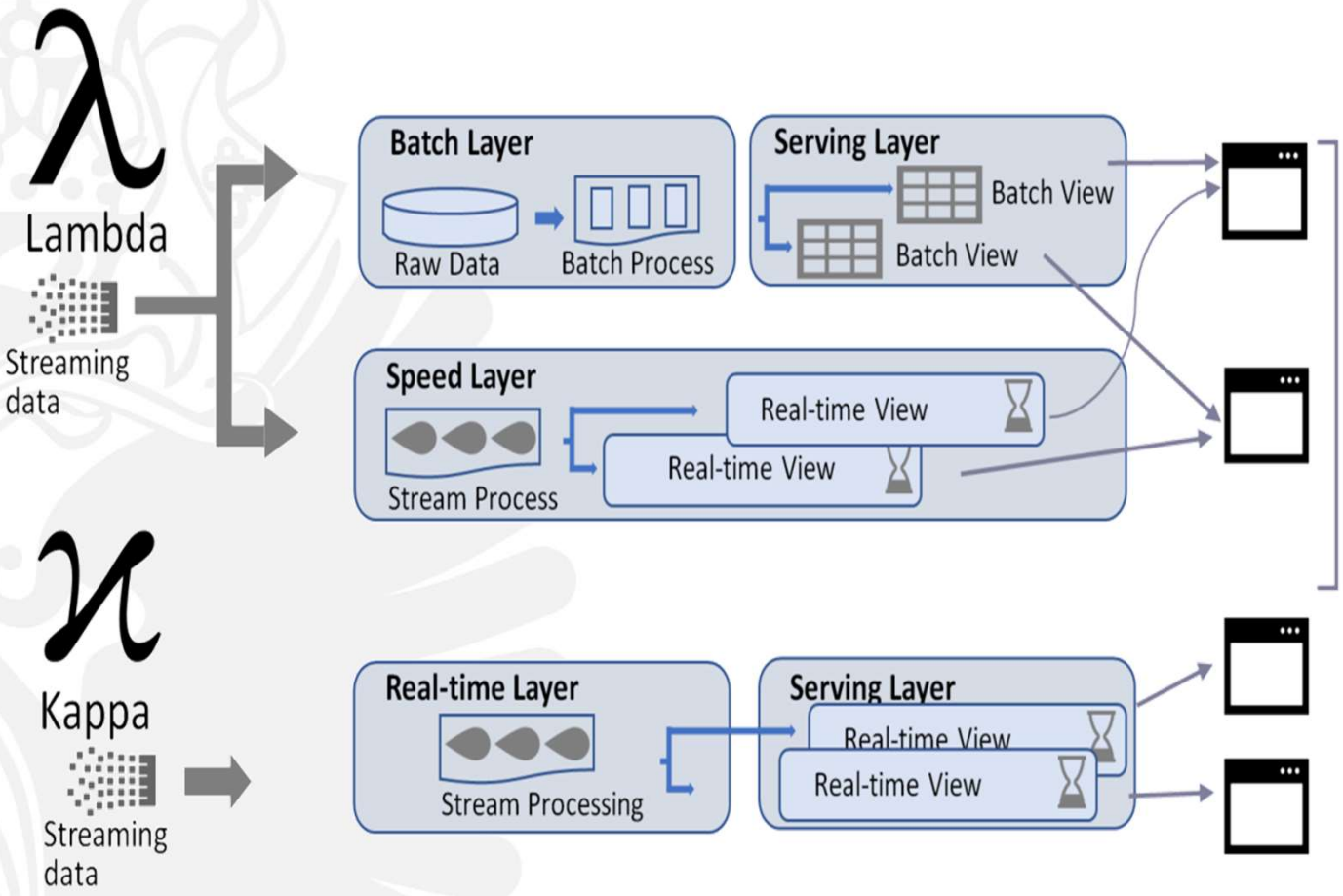
ETL – Extract, Transform, Load: дословно «извлечение, преобразование, загрузка», процесс переноса данных в центральное хранилище

ACID — требования к транзакционным системе (Atomicity – атомарность, Consistency – согласованность, Isolation – изолированность, Durability – стойкость)

BASE — общая для блокчейн и больших данных архитектура хранения, при которой обеспечивается базовая доступность (basic availability), гибкое состояние (soft state), согласованность в конечном счете (eventual consistency);

DAG — Directed Acyclic Graph – ориентированный направленный граф. Структура часто используемая для вычислительных задач из-за способности к топологической сортировки, осуществляемой за конечное время;

ОРГАНИЗАЦИЯ ХРАНЕНИЯ ДАННЫХ



ПРОБЛЕМЫ ЦЕНТРАЛИЗОВАННОЙ ПЛАТФОРМЫ ДАННЫХ

1. *Анализ в реальном* времени и *дорогостоящие* инфраструктуры Big Data
2. Постоянное *появление новых источников* данных
3. Организации стремятся *объединять данные различными способами*, чтобы отразить изменчивость бизнес-среды и требований. Это приводит к увеличению числа преобразований, агрегаций, проекций и срезов данных, что увеличивает время отклика.
4. При реализации архитектур платформ данных специалисты испытывают влияние прошлых поколений архитектур при определении этапов обработки данных.

НОВАЯ ПАРАДИГМА СЕТИ РАСПРЕДЕЛЕННЫХ ДАННЫХ

Для того чтобы *децентрализовать монолитную платформу данных*, необходимо *изменить* наше *представление* о *данных*, их местонахождении и собственности.

Передача данных из доменов в "озеро" или централизованную платформу

Домены

- **Право собственности на наборы данных делегируется от центральной платформы доменам**
- **Домены размещают и поддерживают свои массивы данных в удобной для использования форме**
- **Наборы данных исходного домена должны быть отделены от внутренних наборов данных исходных систем**
- **Обеспечить очистку, подготовку, агрегирование и сопровождение данных, а также использование конвейера данных**
- **Команды, управляющие доменами, предоставляют возможность обработки своих данных другим специалистам организации через API**

Данные

- **Имеют гораздо больший объем, неизменно синхронизируют факты и изменяются реже, чем их системы**
- **Наборы исходных доменных данных являются наиболее фундаментальными и изменяются реже, т.к. бизнес-факты меняются не так часто**
- **Наборы исходных доменных данных являются сырыми данными на момент создания, они не адаптированы и не смоделированы для конкретного потребителя**

!

Должен быть реализован безопасный и управляемый глобальный контроль доступа к массивам данных

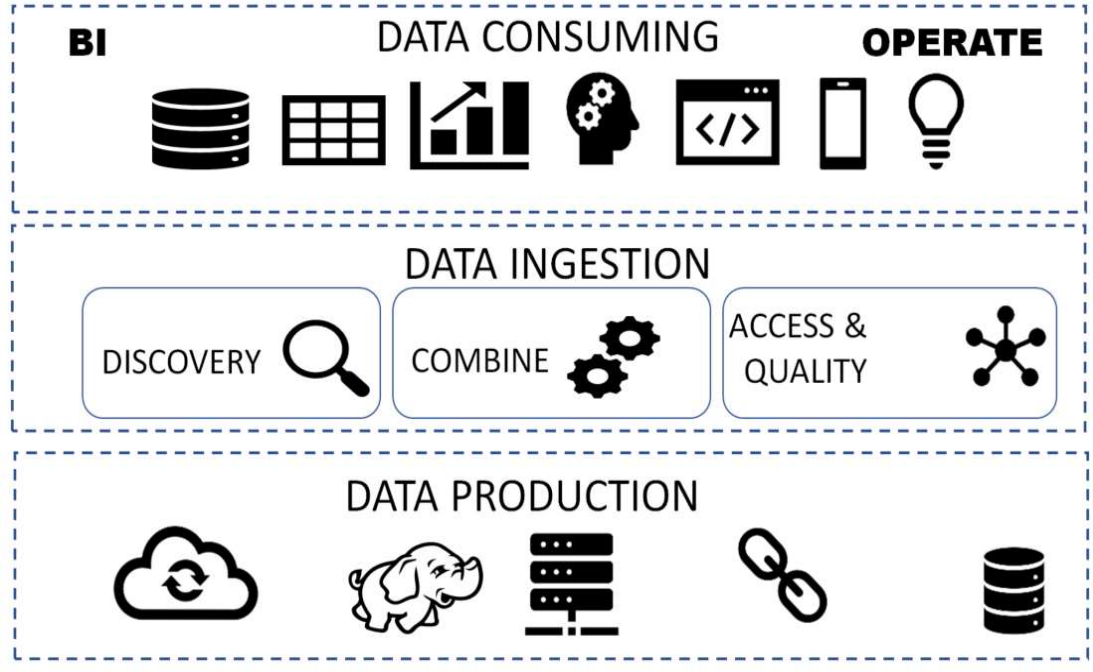
Для обеспечения быстрого поиска необходимых данных должен быть реализован каталог данных, содержащий метаинформацию обо всех данных

ЧТО НАДО СДЕЛАТЬ?

- Необходимо централизовано найти данные, которые необходимо обработать
- Обработать их там, где они лежат
- Вставить их в ту инфраструктуру, которая нам требуется

Не работать с системой «в целом» иметь возможность где-то локально что-то изменить и получать только результат

ВИРТУАЛИЗАЦИЯ ДАННЫХ



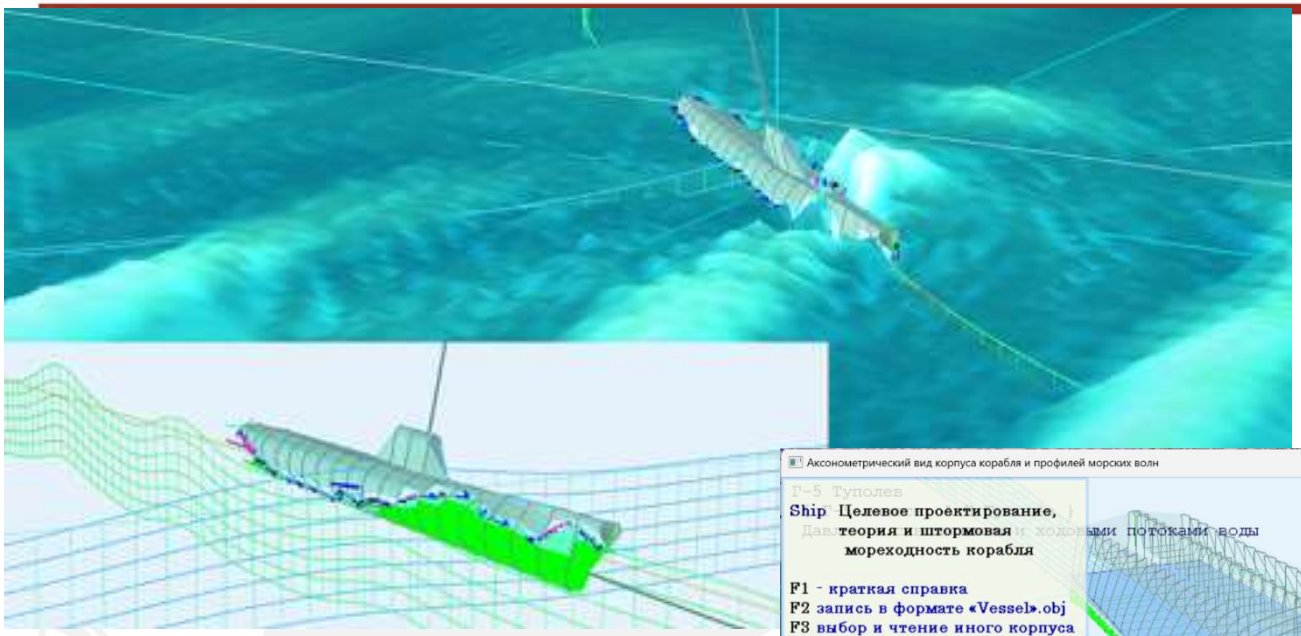
Это обеспечивает *доступ к данным* из большого количества распределенных источников и различных форматов, при этом пользователям не требуется знать, где они хранятся.

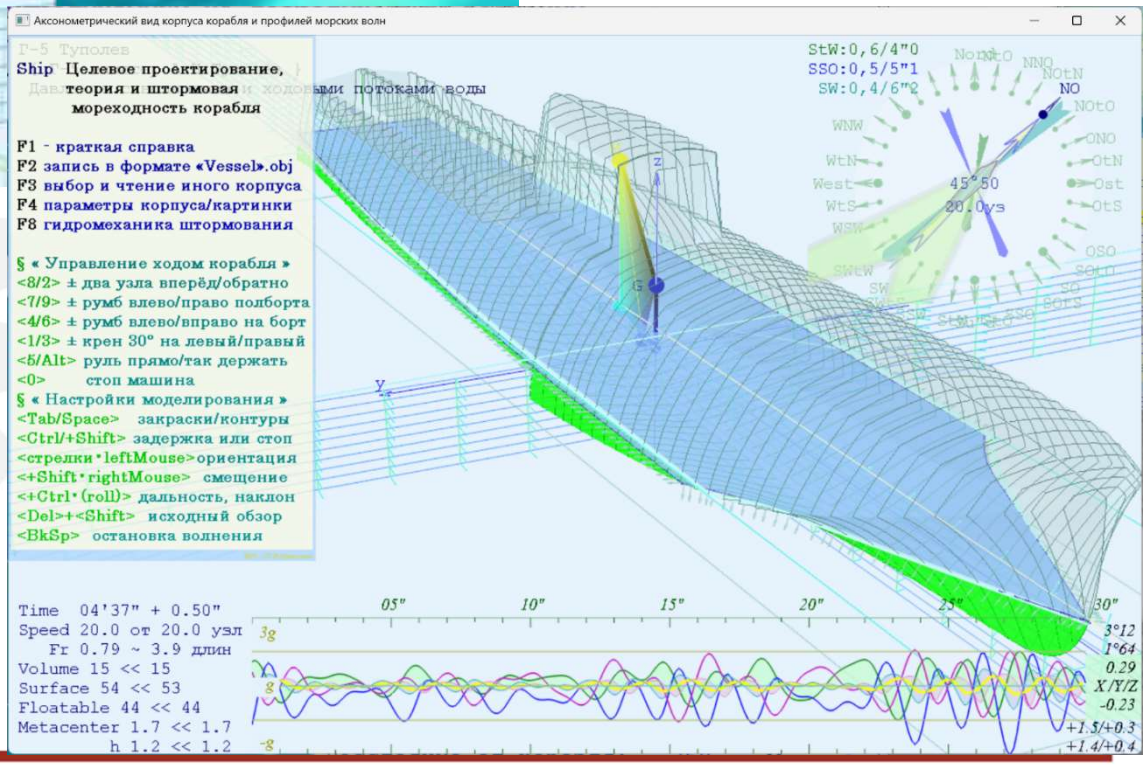
При этом *отпадает* необходимость в *перемещении данных* или *выделении ресурсов* для их хранения.

Помимо повышения эффективности и ускорения доступа к данным, виртуализация данных может дать необходимую основу для выполнения требований по *управлению данными*.

Санкт-Петербургский государственный университет
www.spbu.ru

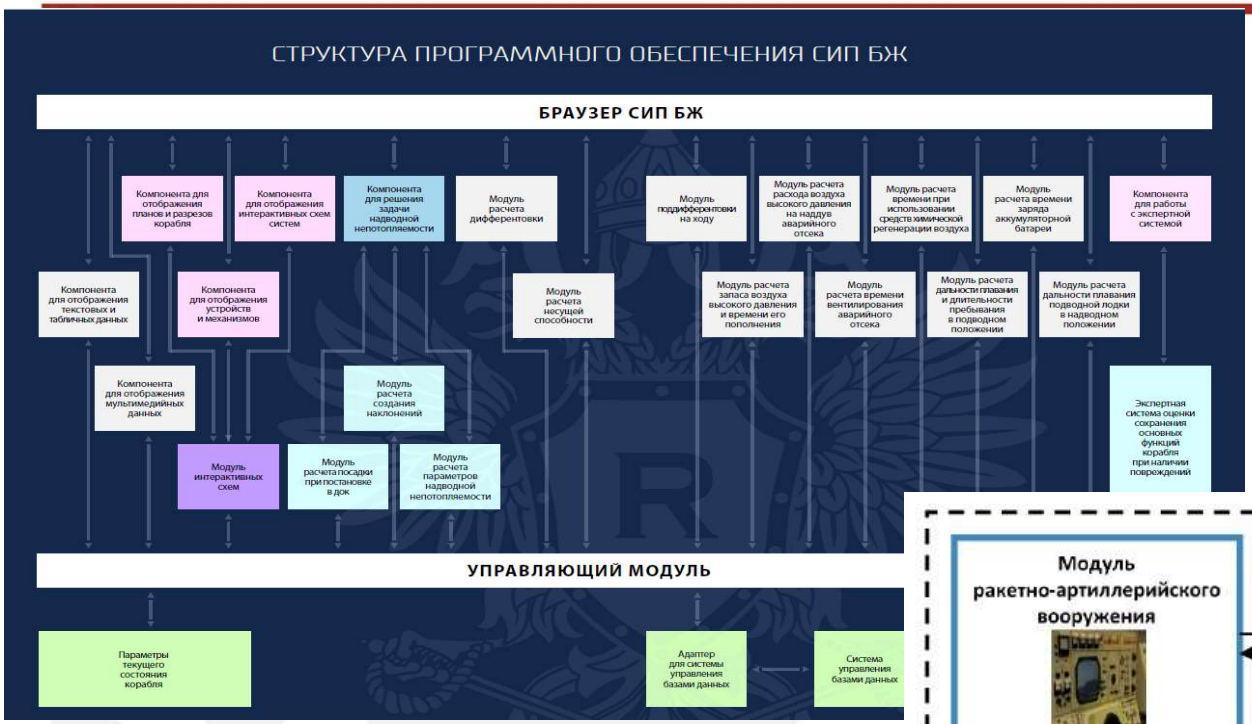
ВИРТУАЛЬНЫЙ ПОЛИГОН

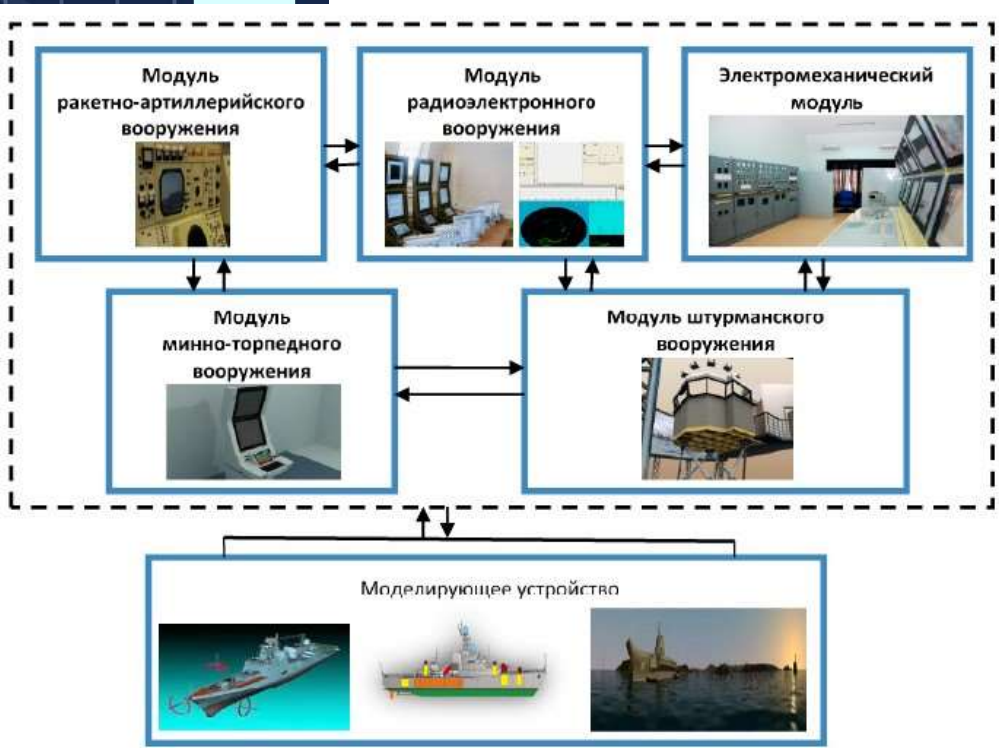


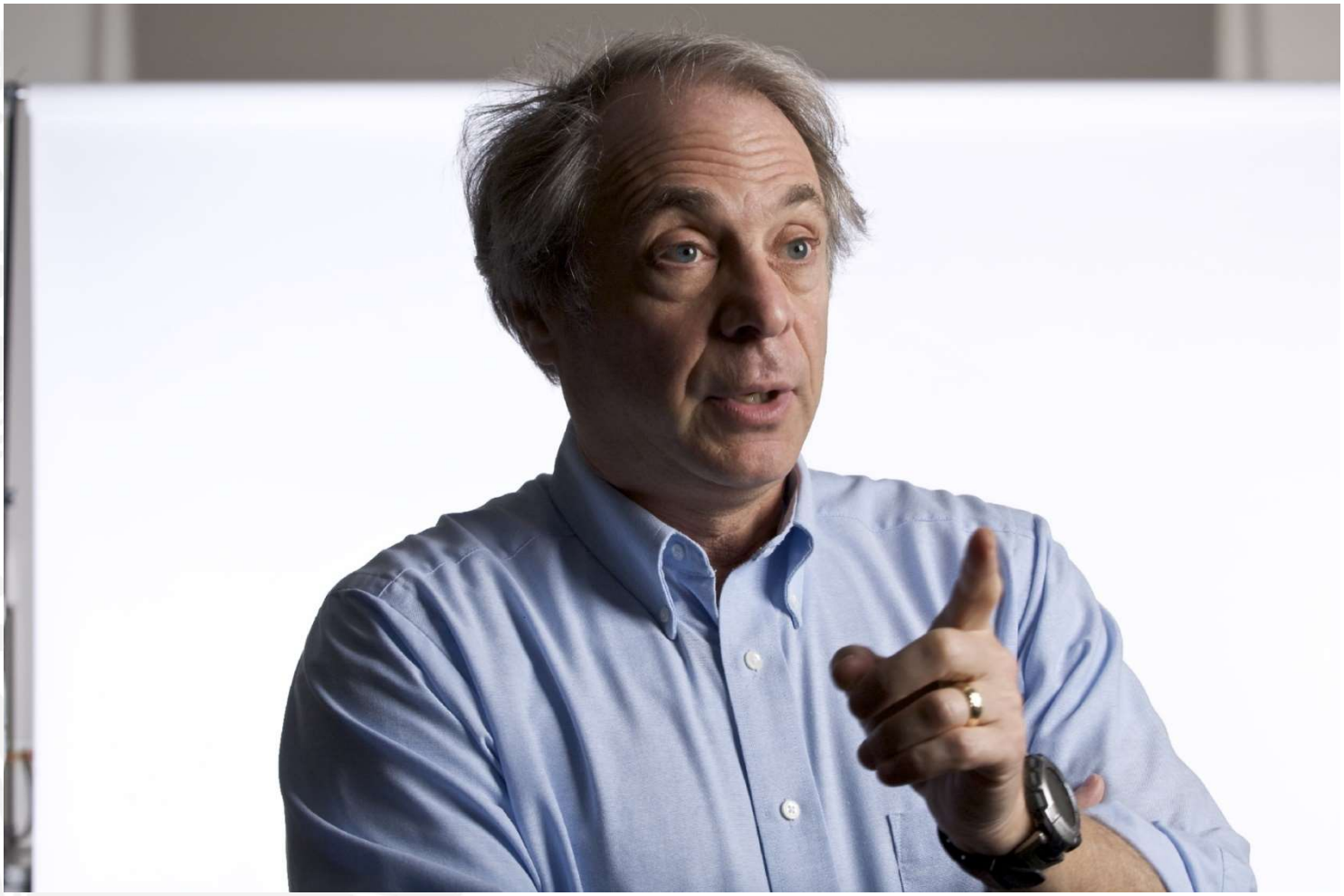


Санкт-Петербургский государственный университет
www.spbu.ru

КОМПЛЕКСНЫЙ ТРЕНАЖЕР







The difference between having power and using it

Steve Wallach

СПАСИБО ЗА ВНИМАНИЕ

Санкт-Петербургский
государственный университет
spbu.ru