Коммутаторы
Альтернативные методы построения процессоров
Видеокарты
Пиринговые сети
Грид-системы

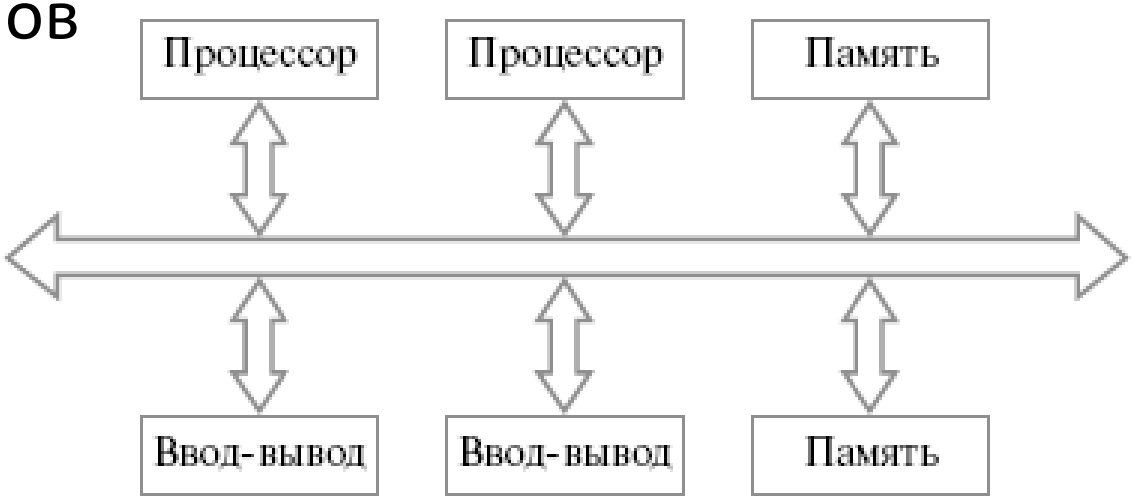
Лекция 2

8 октября 2018

Коммутаторы

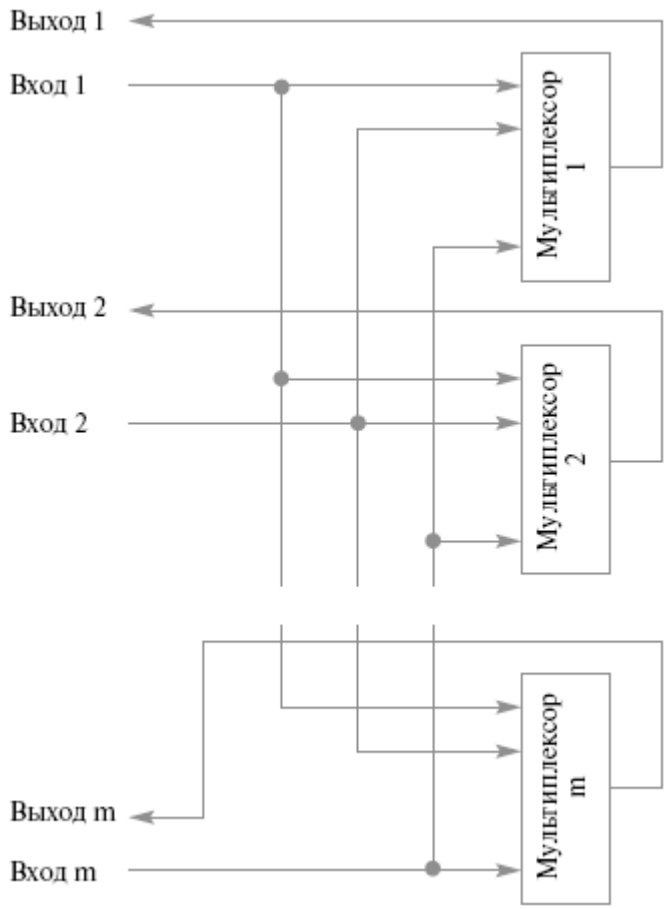
Простые коммутаторы

- С временным разделением (*с помощью алгоритмов арбитража*)
  - Назначение устройству приоритета (статического или динамического)
  - Использование очереди запросов FIFO
  - Выделение фиксированных временных интервалов



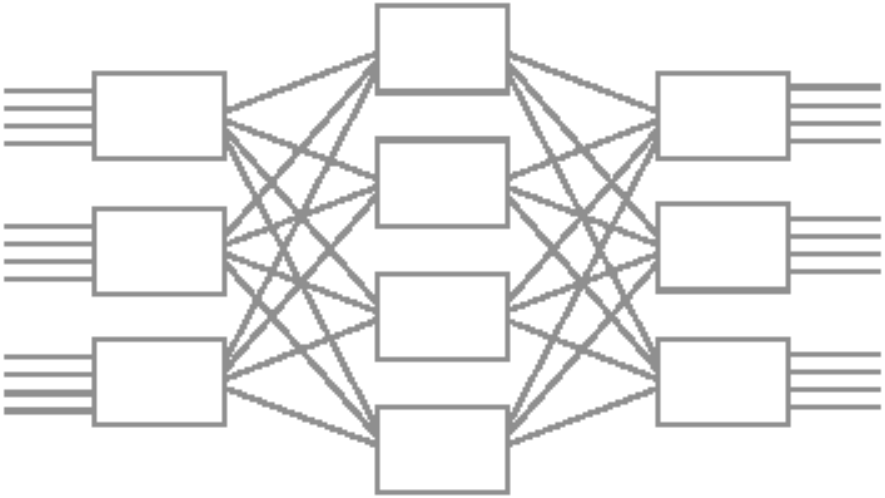
Простые коммутаторы

- С пространственным разделением



Составные коммутаторы. Коммутатор Клоза

Может быть построен в качестве альтернативы для прямоугольного коммутатора с (m x d) входами и (m x d) выходами. Он формируется из трех каскадов коммутаторов: m коммутаторов (d x d) во входном каскаде, m коммутаторов (d x d) в выходном и d промежуточных коммутаторов (m x m).



Соединения внутри коммутатора устроены следующим образом:

- j-й выход i-ого коммутатора входного каскада соединен с i-ым входом j-ого промежуточного коммутатора;
- j-й вход k-ого коммутатора выходного каскада соединен с k-ым выходом j-ого промежуточного коммутатора.

Данный тип составных коммутаторов позволяет соединять любой вход с любым выходом, однако при установленных соединениях добавление нового соединения может потребовать разрыва и переустановления всех соединений.

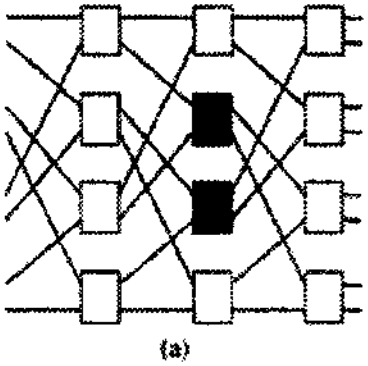
Составные коммутаторы. Баньян-сети

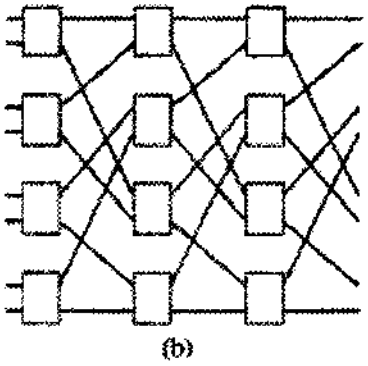
Коммутаторы этого типа строятся на базе прямоугольных коммутаторов таким образом, что существует только один путь от каждого входа к каждому выходу.

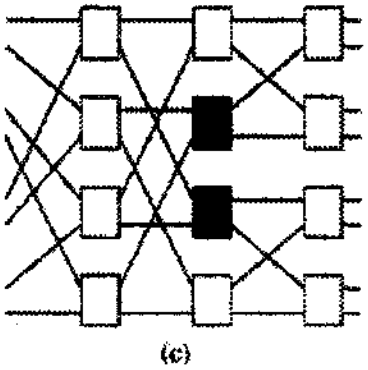
Виды баньян-сетей

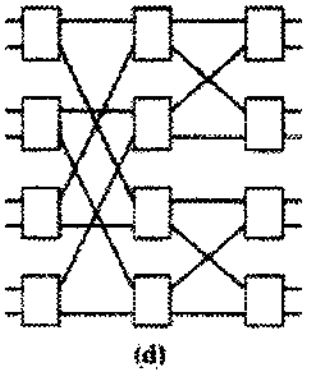
Наиболее важной разновидностью баньян-сетей является дельта-сеть.
Она формируется из прямоугольных коммутаторов (A x B) и представляет собой n-каскадный коммутатор с An входами и Bn выходами. Составляющие коммутаторы соединены так, что для соединения любого входа и выхода образуется единственный путь одинаковой для всех пар входов и выходов длины.

a) смешанные (Омега) сеть;
b) реверсная смешанная сеть;
c) особо чувствительная Баньян-сеть;
d) обыкновенная сеть;









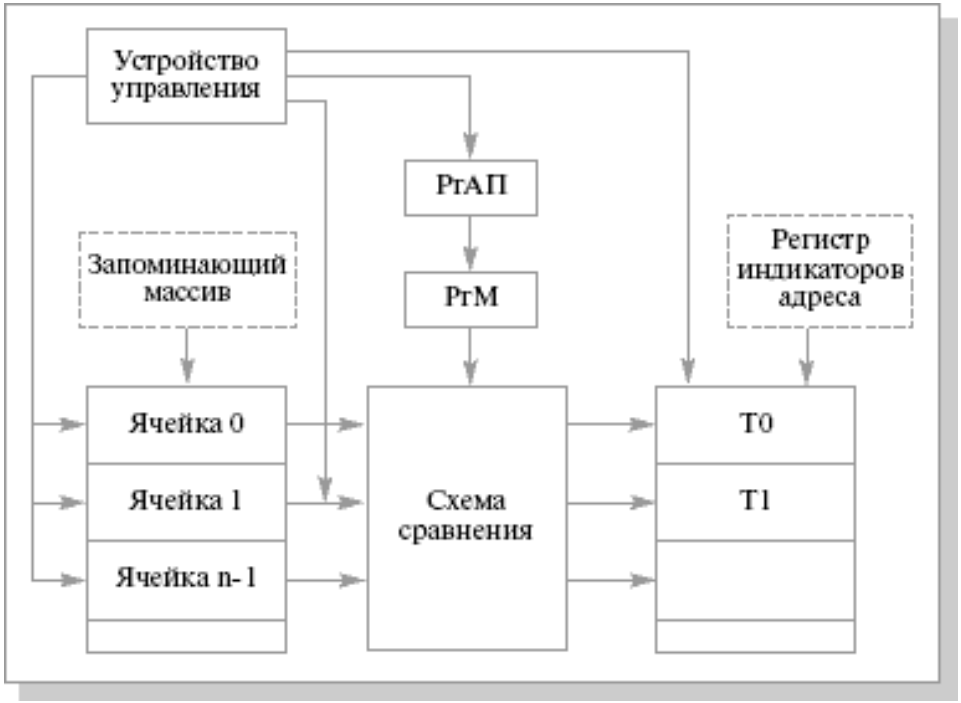
Альтернативные методы построения процессоров

Ассоциативные процессоры

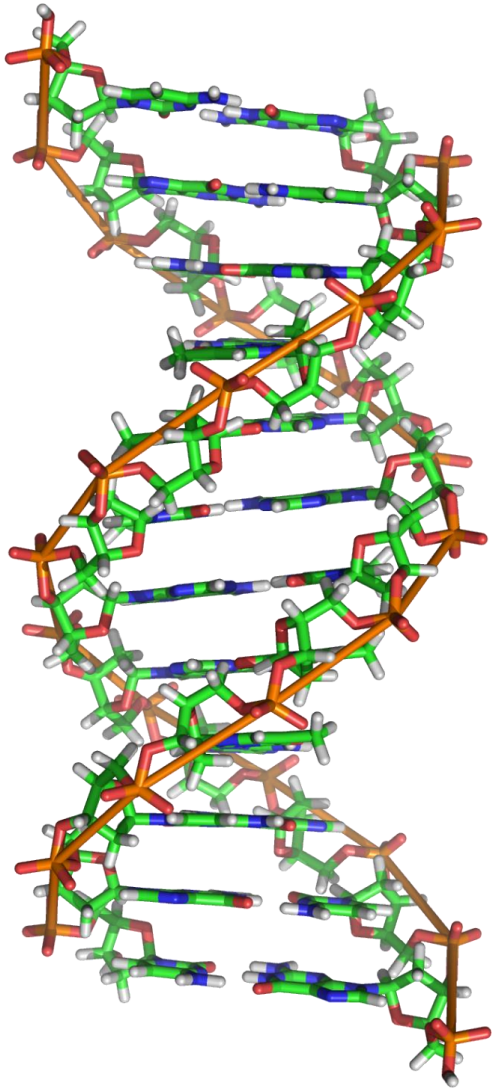
Относятся к классу SIMD

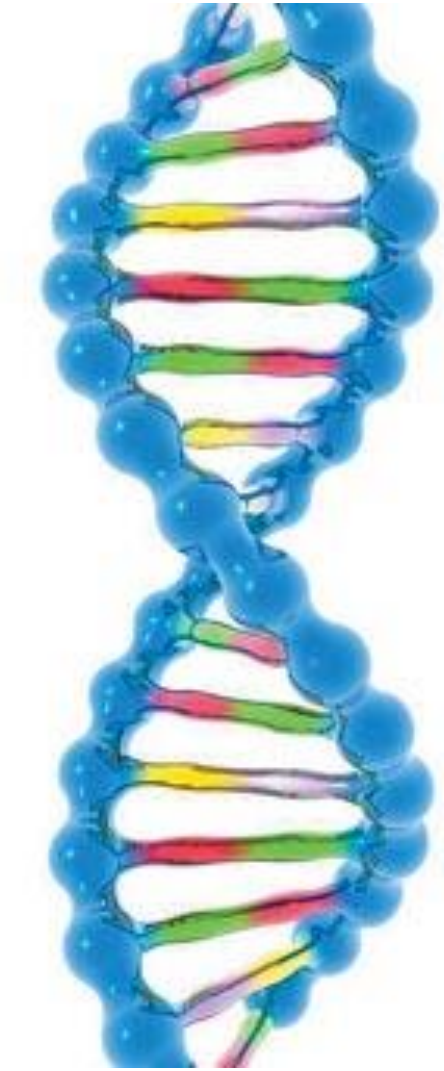
Адресация памяти
по содержимому

АЗУ – Ассоциативное
Запоминающее
Устройство

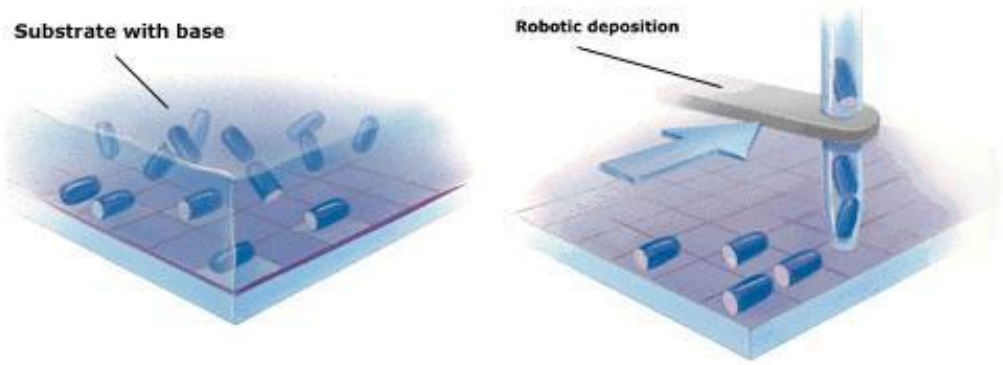


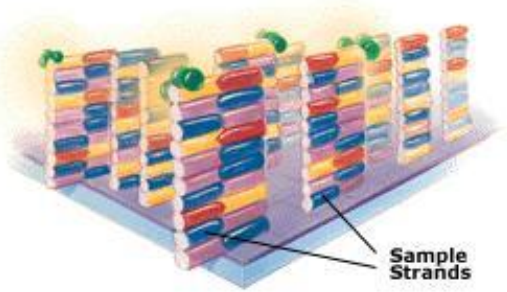
ДНК-процессоры

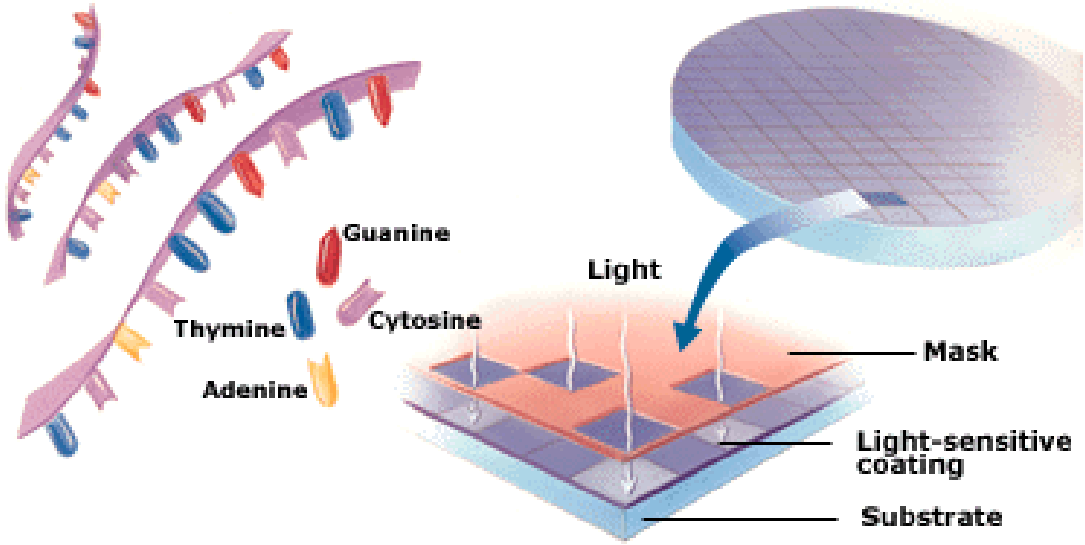




ДНК-процессоры







ДНК-процессоры

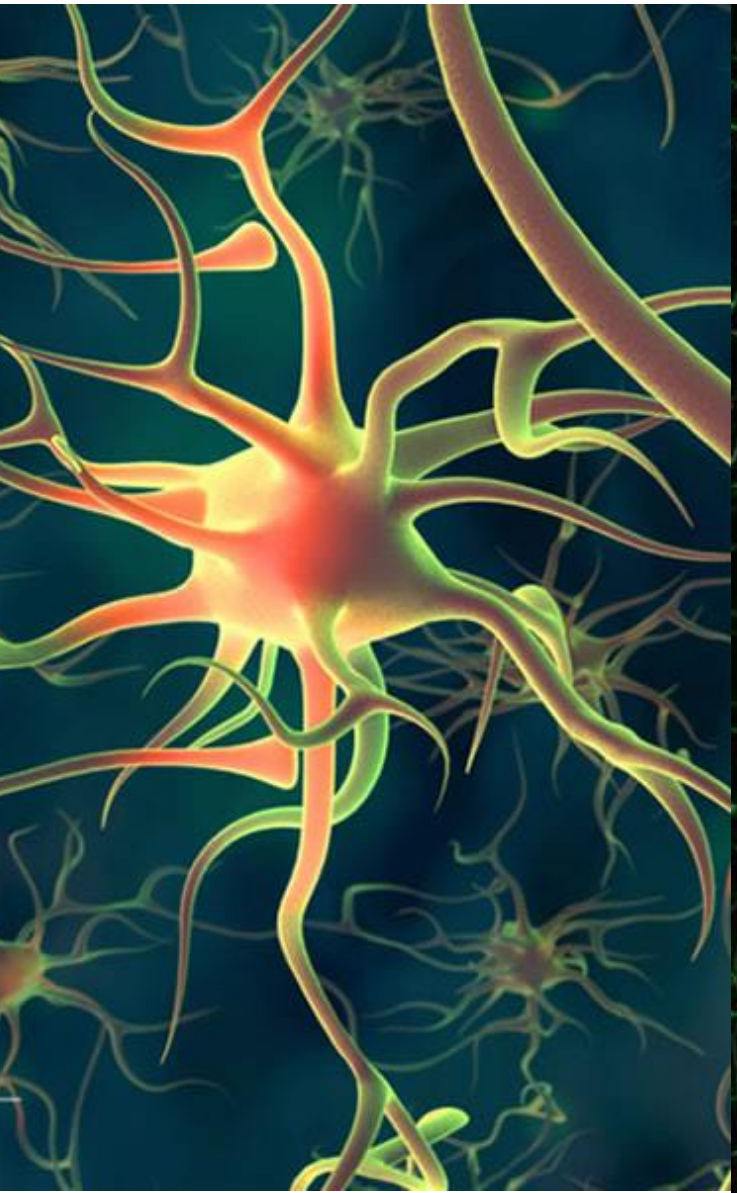
Достоинства:

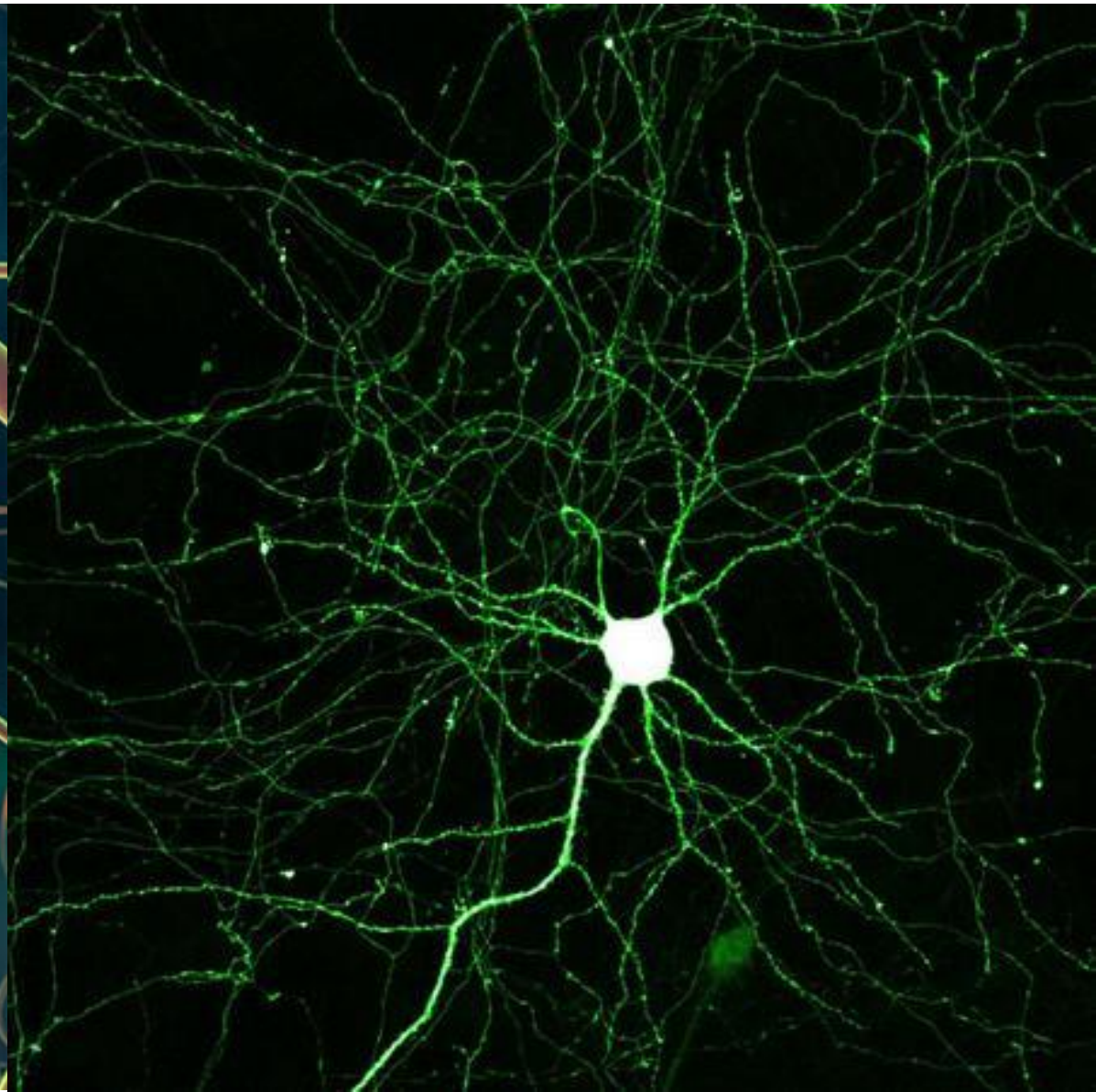
- более простая технология изготовления, не требующая для своей реализации столь жестких условий, как при производстве полупроводников;
- использование не бинарного, а тернарного кода (информация кодируется тройками нуклеотидов), что позволит за меньшее количество шагов перебрать большее число вариантов при анализе сложных систем;
- потенциально исключительно высокая производительность, которая может составлять до 10^{14} операций в секунду за счет одновременного вступления в реакцию триллионов молекул ДНК;
- возможность хранить данные с плотностью, в триллионы раз превышающей показатели оптических дисков;
- исключительно низкое энергопотребление.

Недостатки:

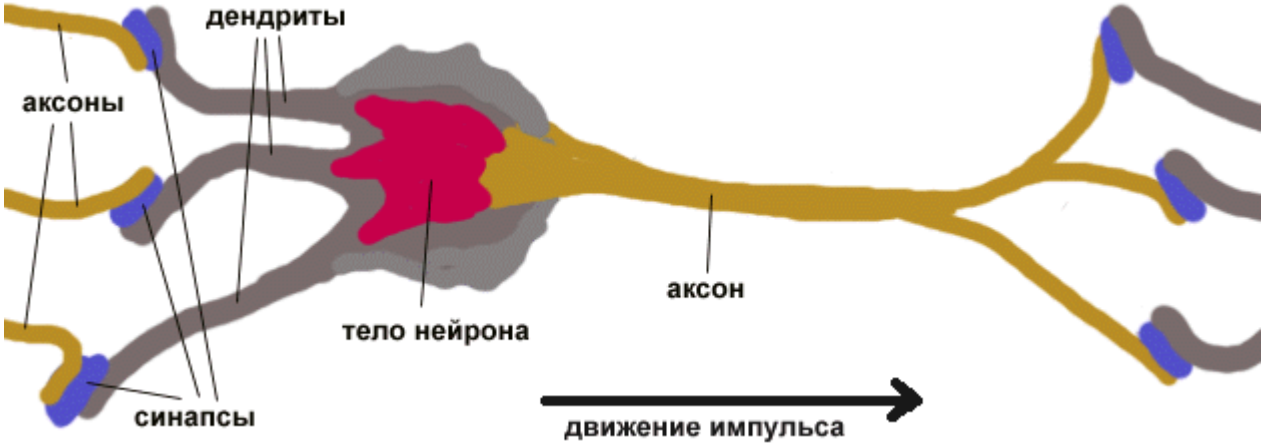
- сложность со считыванием результатов – современные способы определения кодирующей последовательности несовершенны, сложны, трудоемки и дороги;
- низкая точность вычислений, связанная с возникновением мутаций, прилипанием молекул к стенкам сосудов и т.д.;
- невозможность длительного хранения результатов вычислений в связи с распадом ДНК в течение времени.

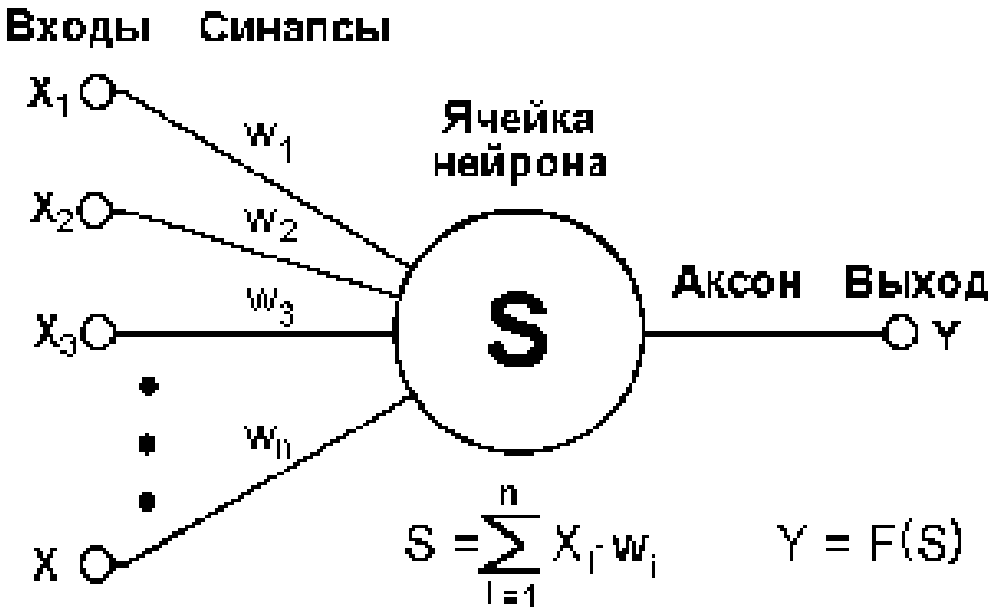
Нейронные сети



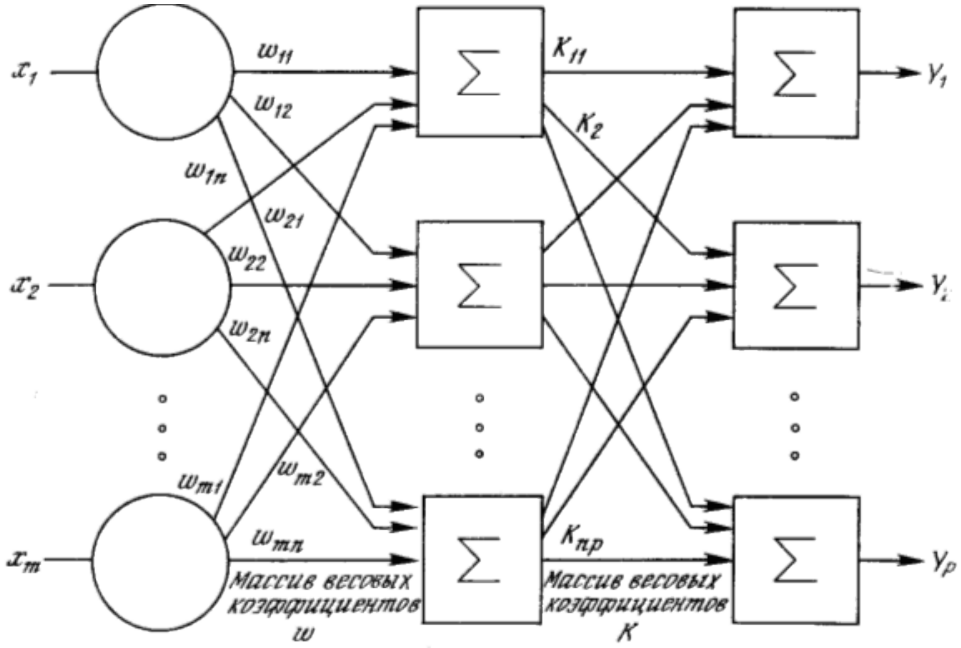


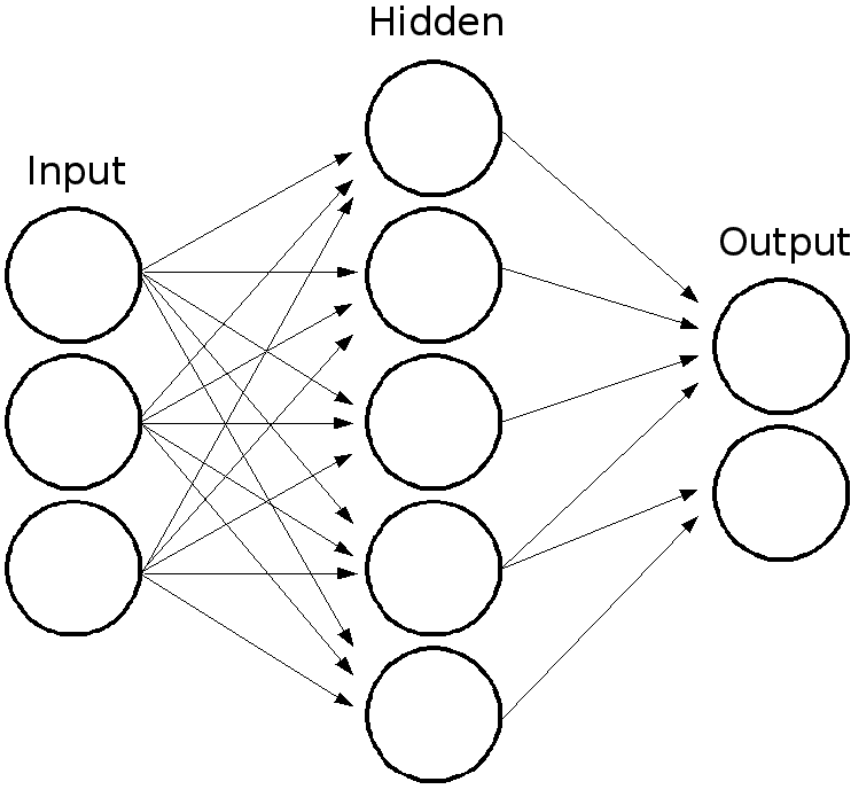
Нейрон


дендриты
аксоны
аксон
тело нейрона
синапсы
движение импульса

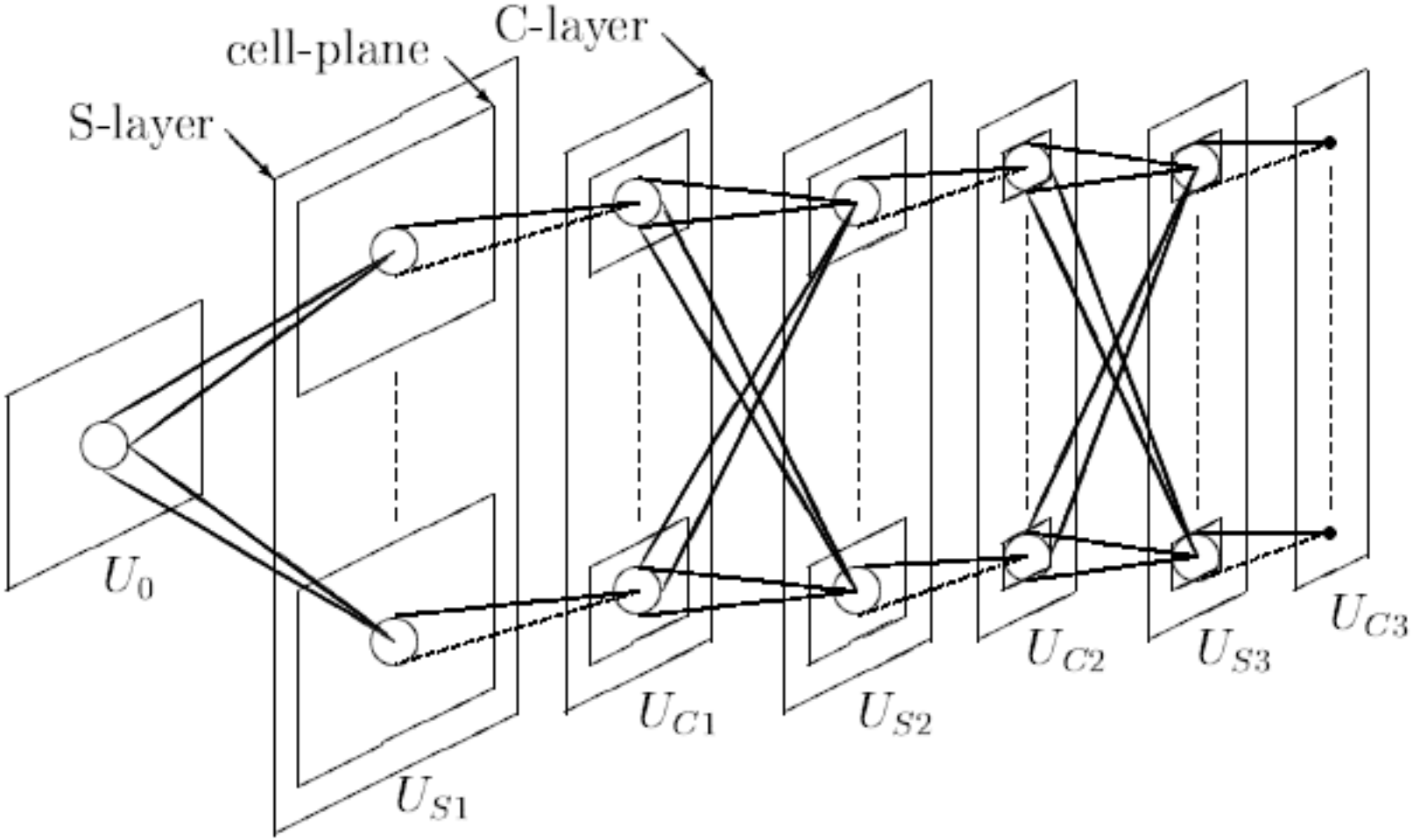

Входы
Синапсы
X1
X2
X3
X
w1
w2
w3
wn
Ячейка нейрона
S
Аксон
Выход
Y
S = Σ Xi·wi
i=1
n
Y = F(S)

Искусственные нейронные сети





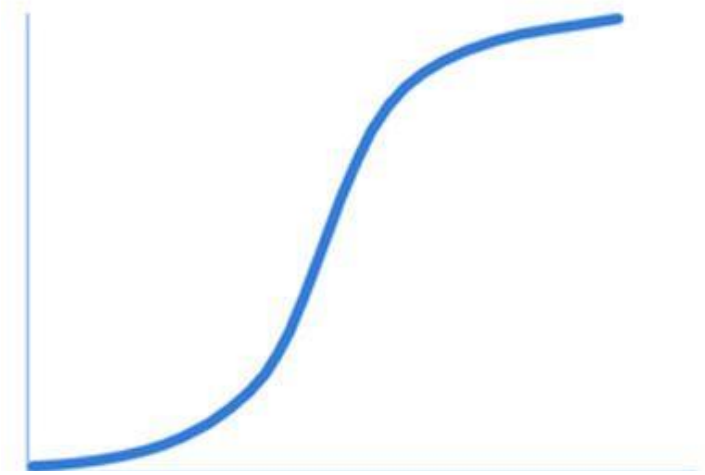
Искусственные нейронные сети

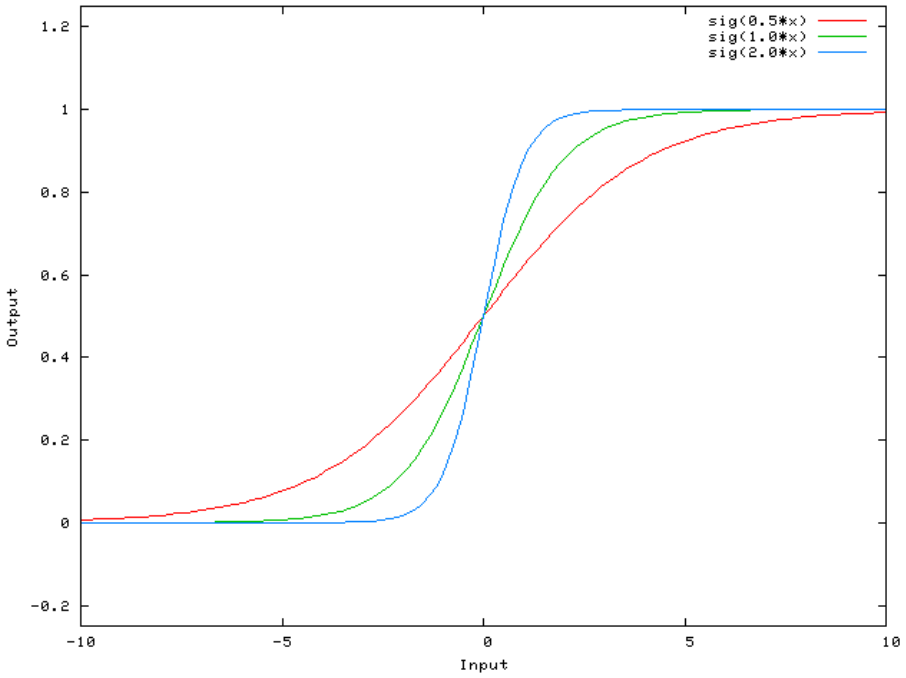


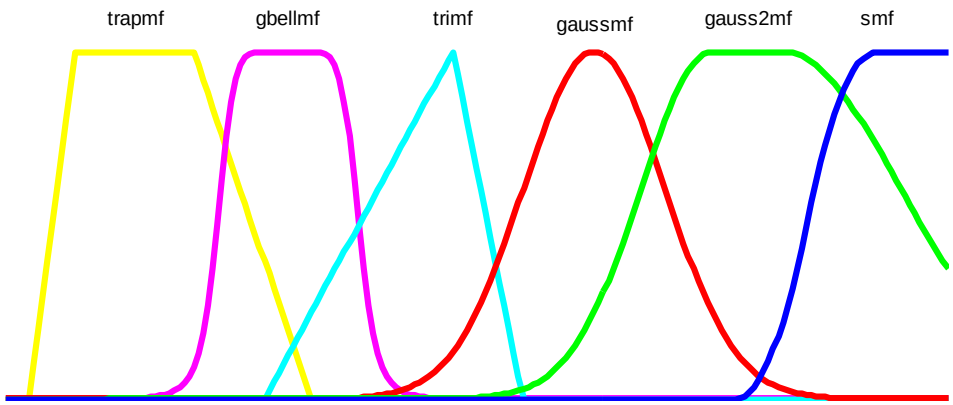
Искусственные нейронные сети

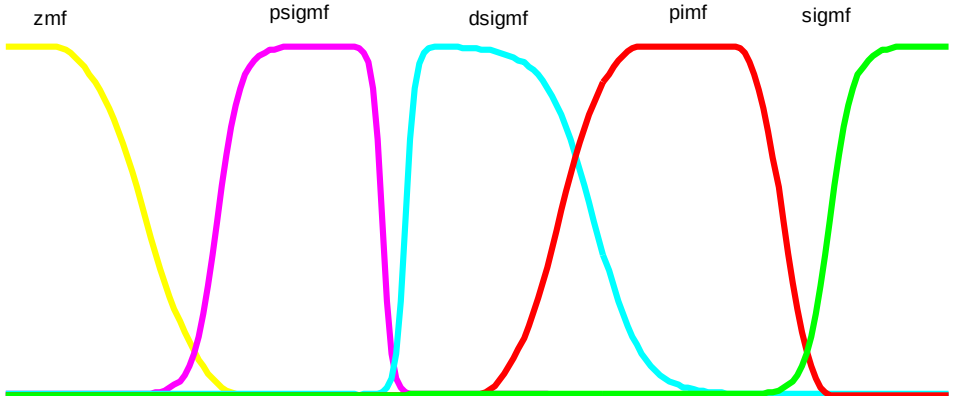
- **Нейроны группируются в последовательность слоёв**;
входные сигналы поступают на первый слой и последовательно проходят через все слои до последнего, выходного слоя сети. Но бывают и рекуррентные (с циклами) структуры, обеспечивающие циркуляцию некоторого набора внутренних сигналов.
- **Искусственная нейросеть, может обучаться**:
она содержит внутренние адаптивные параметры нейронов и своей структуры, и, меняя их, может менять свое поведение, добиваясь улучшения точности решения некоторой задачи.
- Обучение или настройка нейронной сети: при переходе к новой задаче не нужно заново программировать алгоритм - можно просто взять универсальный нейросетевой инструмент и в нём создать и обучить нейросеть.
- Нейронная сеть обучается решению задачи на некотором наборе ситуаций, каждая из которых описывает значения входных сигналов нейросети и требуемый при этих входных сигналах ответ.
- Кроме **обучения с учителем** (на основе знаний об известных эталонных ответах для некоторого набора ситуаций) возможно и **обучение без учителя** − при этом происходит анализ описаний ситуаций и ищутся те или иные тенденции (например, похожие ситуации объединяются в группы так, чтобы данные внутри группы были больше похожи друг на друга, чем на данные из другой группы).
- Нейронная сеть способна обучаться решению задач, для которых не существует формализованных, быстрых или работающих с приемлемой точностью теоретических или эмпирических алгоритмов.
- **Структура нейросети может быть адаптирована к задаче**.

Процессоры с нечеткой логикой







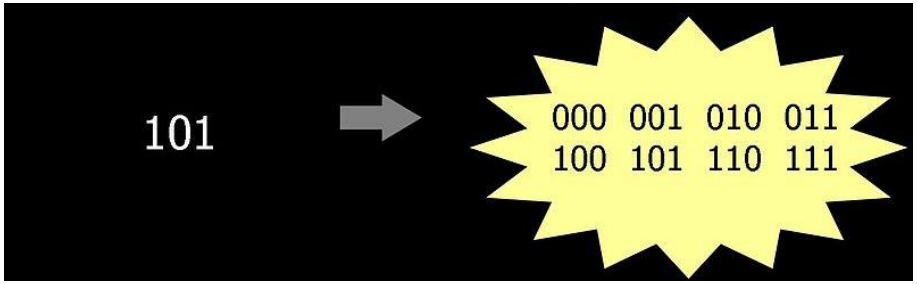


Процессоры с нечеткой логикой

Квантовые компьютеры

Квантовый компьютер — вычислительное устройство, которое использует явления квантовой суперпозиции и квантовой запутанности для передачи и обработки данных.

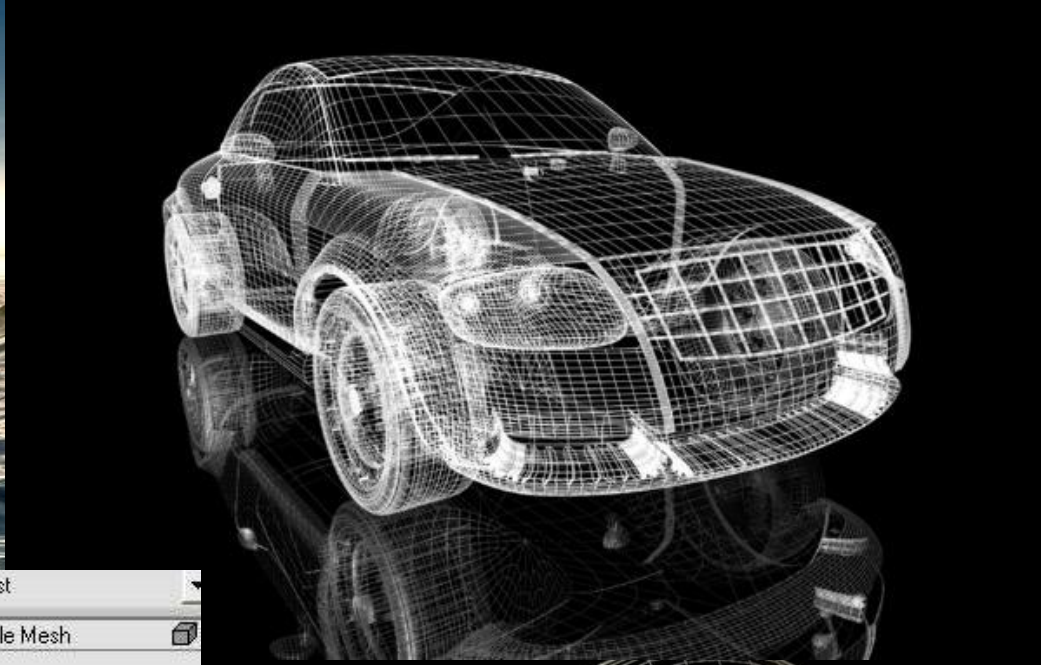
Кубит (q-bit) – квантовый разряд или наименьший элемент для хранения информации.

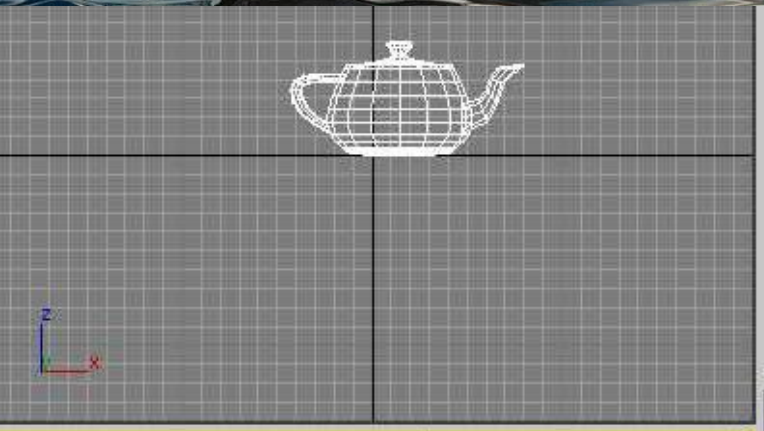


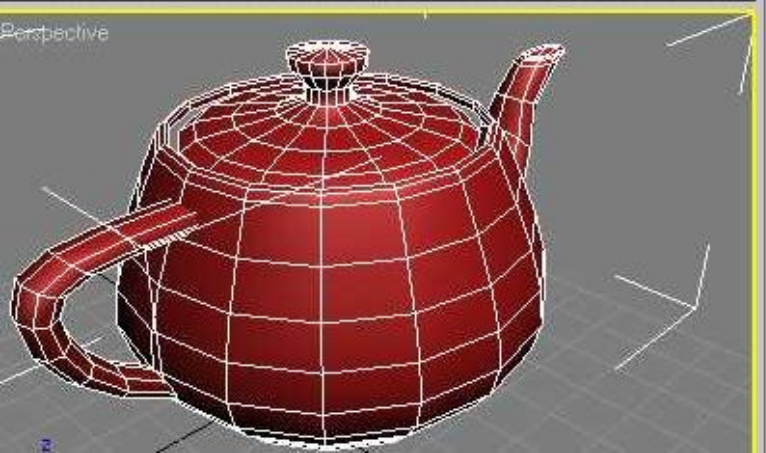
Вычисления заключаются в **унитарных преобразованиях** на **системе кубитов**.

Разработка квантовых компьютеров ведется такими компаниями как Google, Microsoft, IBM и др.

Видеокарты







Modifier List

Editable Mesh

Selection

By Vertex

Ignore Backfacing

Ignore Visible Edges

Planar Thresh: 45,0

Show Normals

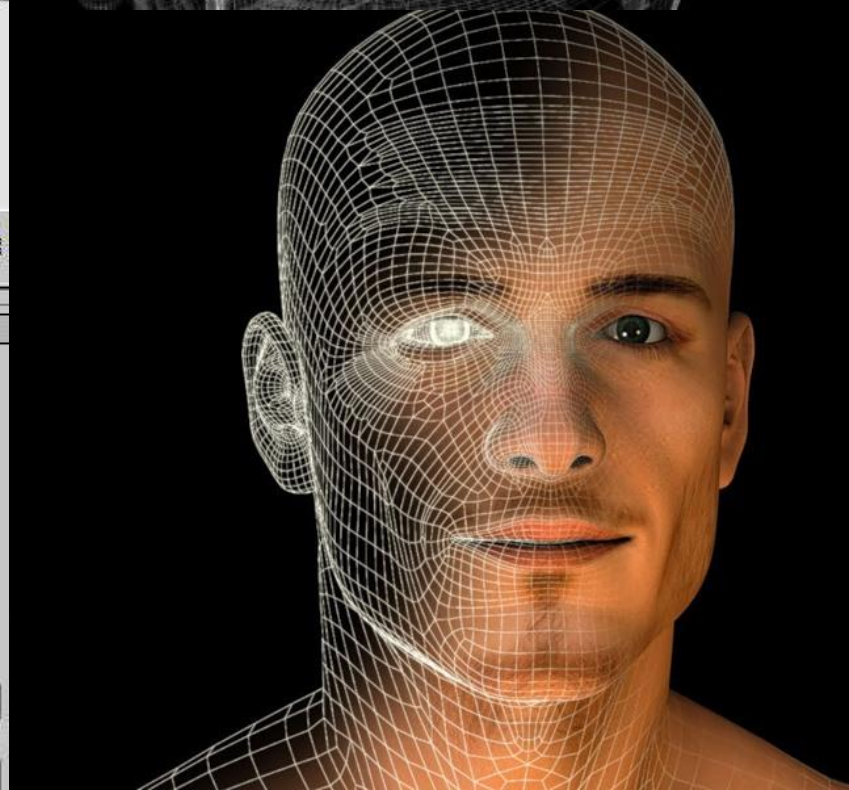
Scale: 20,0

Delete Isolated Vertices

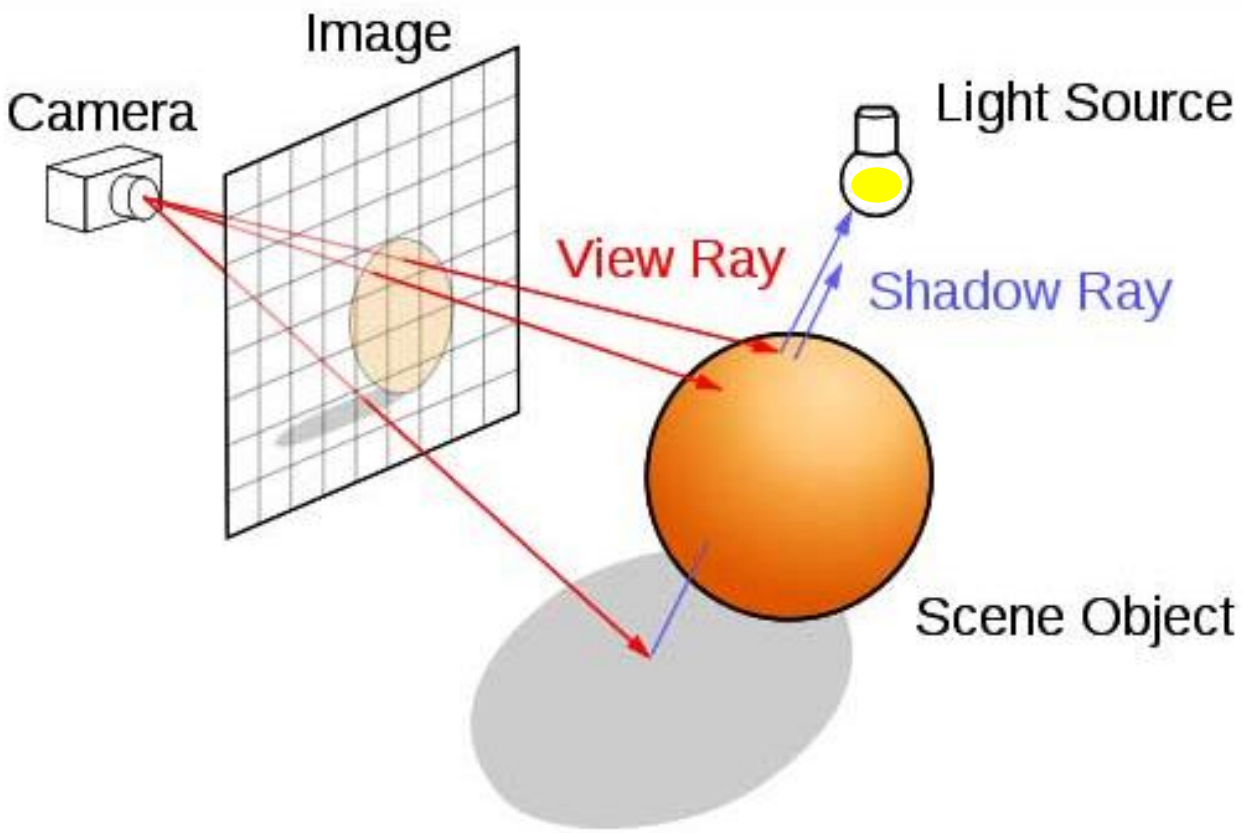
Hide | Unhide All

Named Selections:

Copy | Paste



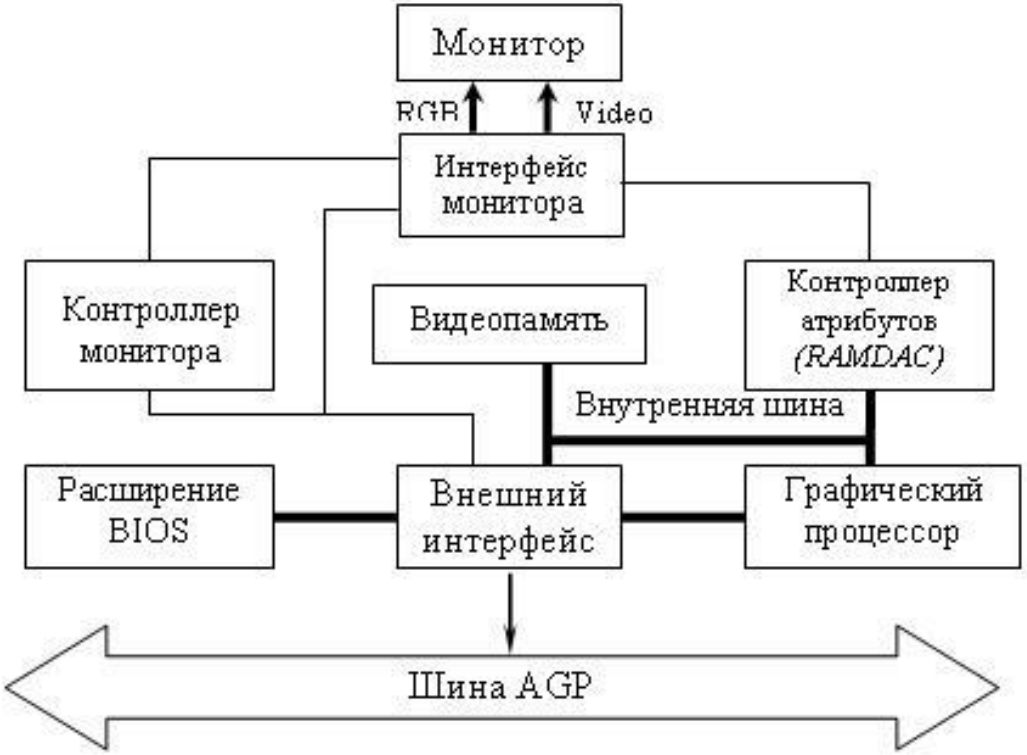
Трассировка лучей



Видеокарта

Схема видеокарты

- Графический процессор
- Видеоконтроллер
- Видео-ПЗУ
- Видеопамять
- Цифро-аналоговый преобразователь (ЦАП) (RAMDAC-*Random Access Memory Digital-to-Analog Converter*)
- Интерфейс вывода на монитор
- Система охлаждения



Виды видеокарт

- **Графика с разделяемой памятью** (*Shared graphics, Shared Memory Architecture*). Видеопамять в виде специализированных ячеек как таковая отсутствует; вместо этого под нужды видеоадаптера динамически выделяется область основной оперативной памяти компьютера. (интегрированные видеокарты, являющиеся частью северного моста).
  Преимущества данного решения — низкая цена и малое энергопотребление. Недостатки — невысокая производительность в 3D-графике и отрицательное влияние на пропускную способность памяти.
- **Дискретная графика** (*Dedicated graphics*). Есть своя видеопамять. Дискретная графика обеспечивает наивысшую производительность в трёхмерной графике. (выполнены в виде отдельных плат)
  Недостатки: более высокая цена и большее энергопотребление.
  Гибридная дискретная графика (*Hybrid graphics*). Комбинация вышеназванных способов, ставшая возможной с появлением шины PCIExpress. Небольшой объём физически распаянной на плате видеопамяти, который может расширяться за счёт использования основной оперативной памяти(ОЗУ).
- **Специализированные видеокарты**. Применяются для высокопроизводительных вычислений.
  Для вывода изображения на монитор не используется.

Характеристики видеокарты

- **ширина шины памяти**, измеряется в битах — количество бит информации, передаваемой за такт. Важный параметр в производительности карты.
- **объём видеопамяти**, измеряется в мегабайтах — объём собственной оперативной памяти видеокарты.
  частота ядра — измеряются в мегагерцах
- **частота памяти** — измеряются в мегагерцах
- **текстурная скорость заполнения** , измеряется в млн. пикселов в секунду, показывает количество выводимой информации в единицу времени
- **пиксельная скорость заполнения**, измеряется в млн. пикселов в секунду, показывает количество выводимой информации в единицу времени.

Технологический процесс изготовления, нм	14
Площадь кристалла, мм²	135
Количество транзисторов, млн	3300
Количество скалярных процессоров (ядер)	640
Количество кластеров обработки графики (GPC)	2
Количество блоков мультипроцессоров (SM)	5
Количество текстурных блоков (TMU)	40
Количество блоков растеризации (ROP)	32
Заполнение сцены, млрд пикс/с	43,3
Заполнение сцены, млрд текс/с	54,1
Разрядность шины видеопамяти, бит	128
Стандарт видеопамяти	GDDR5
Объём видеопамяти, МБ	2048
Пропускная способность шины памяти, ГБ/с	112,0
Интерфейс	PCI Express 3.0 x16
Энергопотребление, Вт	75
Частота ядра, МГц	1354
Частота в режиме Turbo Boost, МГц	1455
Реальная (Номинальная) частота видеопамяти, МГц	1750 (7000)
Производительность FP32, GFLOPS	1733,1
Производительность FP64, GFLOPS	54,1
Производительность FP16, GFLOPS	27,0
Поддержка версий API	Direct3D 12_1, OpenGL 4.5, Vulkan 1.0, OpenCL 2.0
Поддержка версии Shader Model	Shader Model 5.0

Типы видеопамяти

- **FPM DRAM** (Fast Page Mode Dynamic RAM — динамическое ОЗУ с быстрым страничным доступом)
- **VRAM** (Video RAM — видео ОЗУ) — так называемая двухпортовая DRAM
- **WRAM** (Window RAM)
- **EDO DRAM** (Extended Data Out DRAM — динамическое ОЗУ с расширенным временем удержания данных на выходе)
- **SDRAM** (Synchronous Dynamic RAM — синхронное динамическое ОЗУ)
- **DDR SDRAM** (Double Data Rate)
- **SGRAM** (Synchronous Graphics RAM — синхронное графическое ОЗУ)
- **MDRAM** (Multibank DRAM — многобанковое ОЗУ)
- **RDRAM** (RAMBus DRAM) — память, использующая специальный канал передачи данных (Rambus Channel)

Типы шейдеров

- **Вершинные шейдеры (Vertex Shader)**
  Вершинный шейдер оперирует данными, сопоставленными с вершинами многогранников. К таким данным, в частности, относятся координаты вершины в пространстве, текстурные координаты, тангенс-вектор, вектор бинормали, вектор нормали. Вершинный шейдер может быть использован для видового и перспективного преобразования вершин, генерации текстурных координат, расчета освещения и т. д.
- **Геометрические шейдеры (Geometry Shader)**
  Геометрический шейдер, в отличие от вершинного, способен обработать не только одну вершину, но и целый примитив. Это может быть отрезок (две вершины) и треугольник (три вершины), а при наличии информации о смежных вершинах (adjacency) может быть обработано до шести вершин для треугольного примитива. Кроме того, геометрический шейдер способен генерировать примитивы «на лету», не задействуя при этом центральный процессор. Впервые начал использоваться на видеокартах Nvidia серии 8.
- **Пиксельные шейдеры (Pixel Shader)**
  Фрагментный шейдер работает с фрагментами изображения. Под фрагментом изображения в данном случае понимается пиксель, которому поставлен в соответствие некоторый набор атрибутов, таких как цвет, глубина, текстурные координаты. Фрагментный шейдер используется на последней стадии графического конвейера для формирования фрагмента изображения.

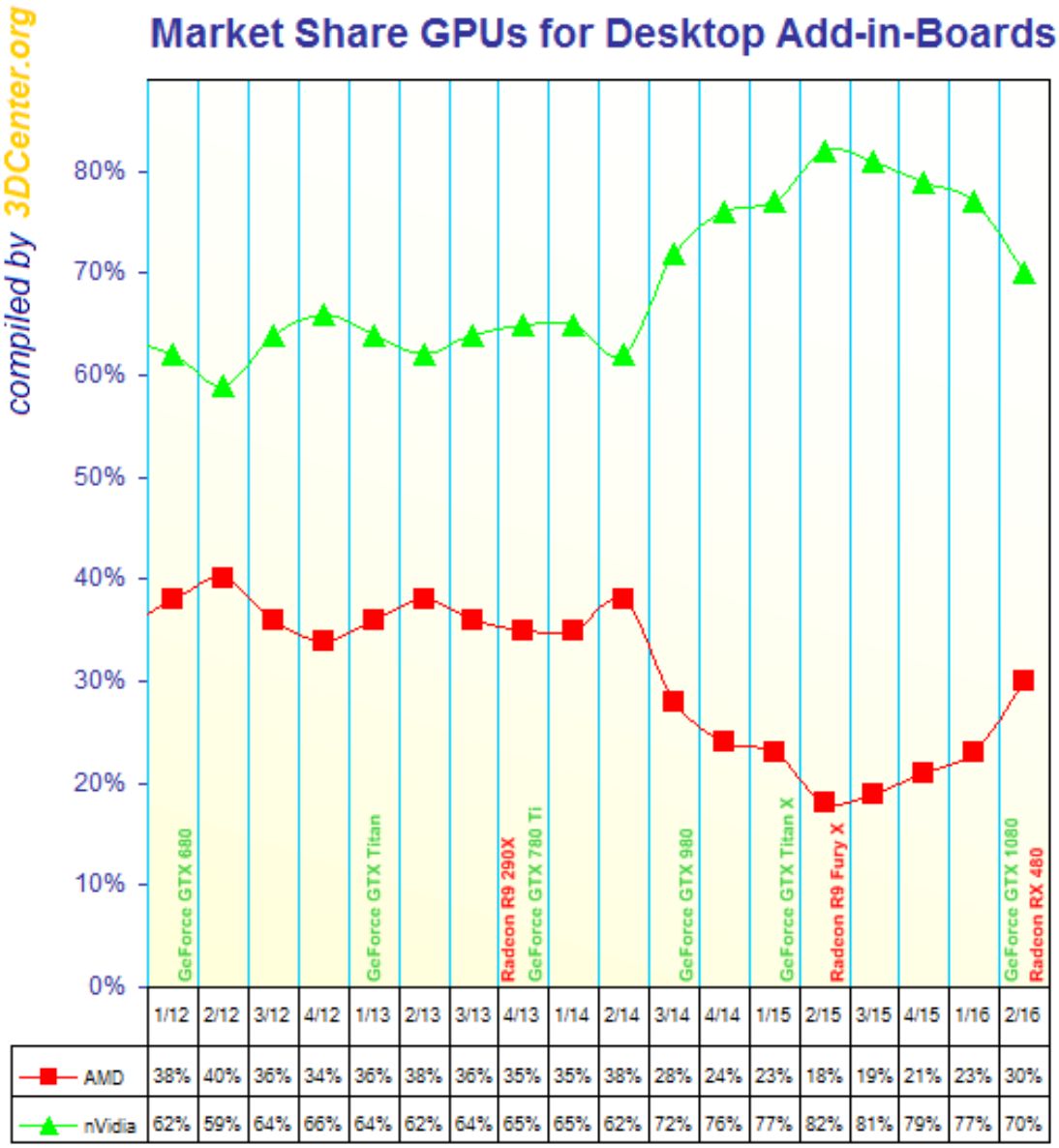
Особенности версий DirectX

DirectX 6.0 — мультитекстурирование
DirectX 7.0 — аппаратная поддержка преобразований, обрезания и освещения
DirectX 8.0 — шейдерная модель 1.1
DirectX 8.1 — пиксельные шейдеры 1.4 и вершинные шейдеры 1.1
DirectX 9.0 — шейдерная модель 2.0
DirectX 9.0b — пиксельные шейдеры 2.0b и вершинные шейдеры 2.0
DirectX 9.0c — шейдерная модель 3.0
DirectX 9.0L — версия DirectX 9.0 для Windows Vista
DirectX 10 — шейдерная модель 4.0 (только Windows Vista, Windows 7, Windows 8)
DirectX 10.1 — шейдерная модель 4.1 (только Windows Vista, Windows 7, Windows 8)
DirectX 11 — шейдерная модель 5.0 (только Windows Vista, Windows 7, Windows 8)
DirectX 11.3/DirectX 12 — шейдерная модель 5.1 (только Windows 10).

Вычисления
на графических картах

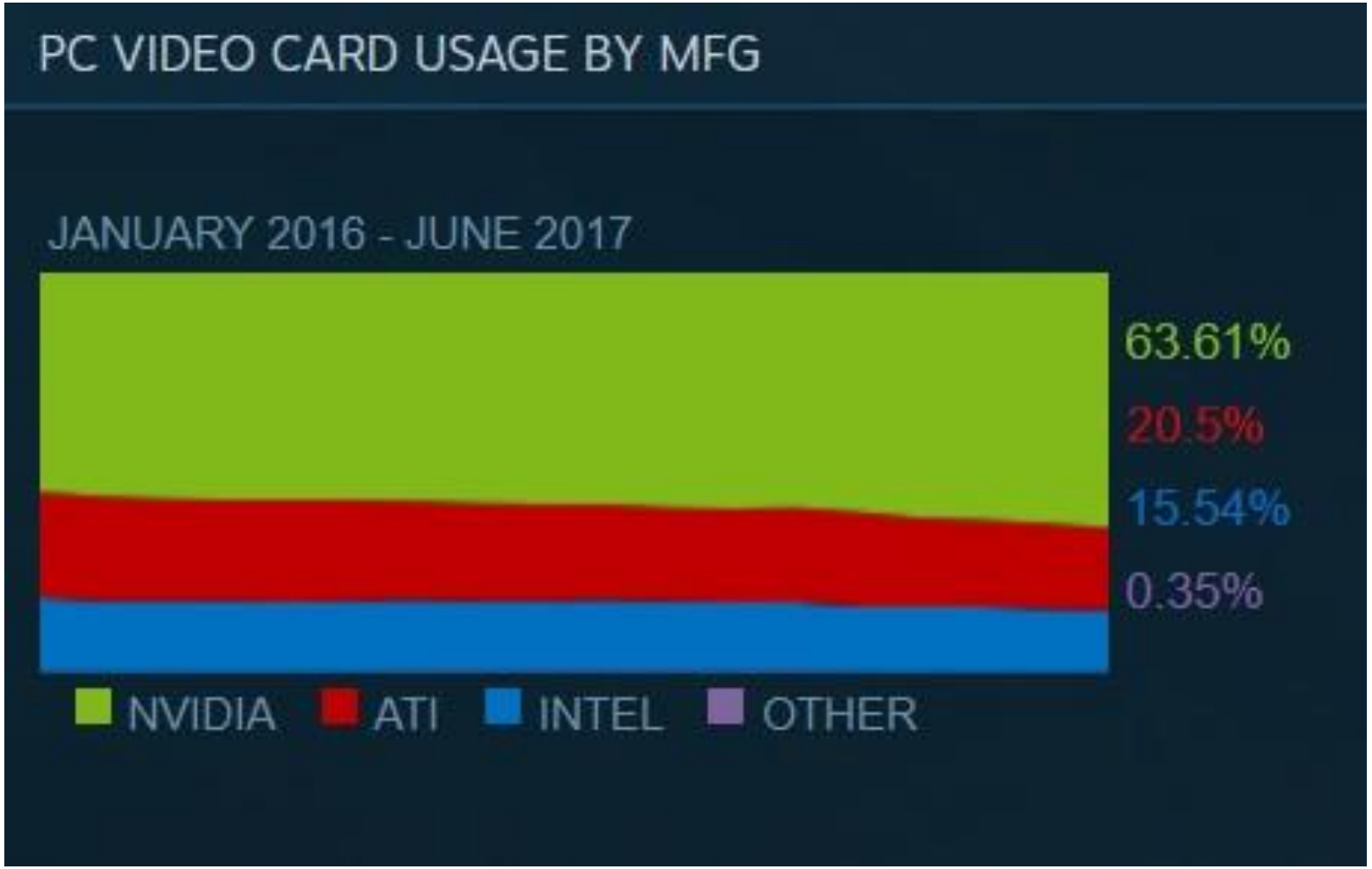
Видеокарты для вычислений

Доля рынка видеокарт AMD и NVIDIA

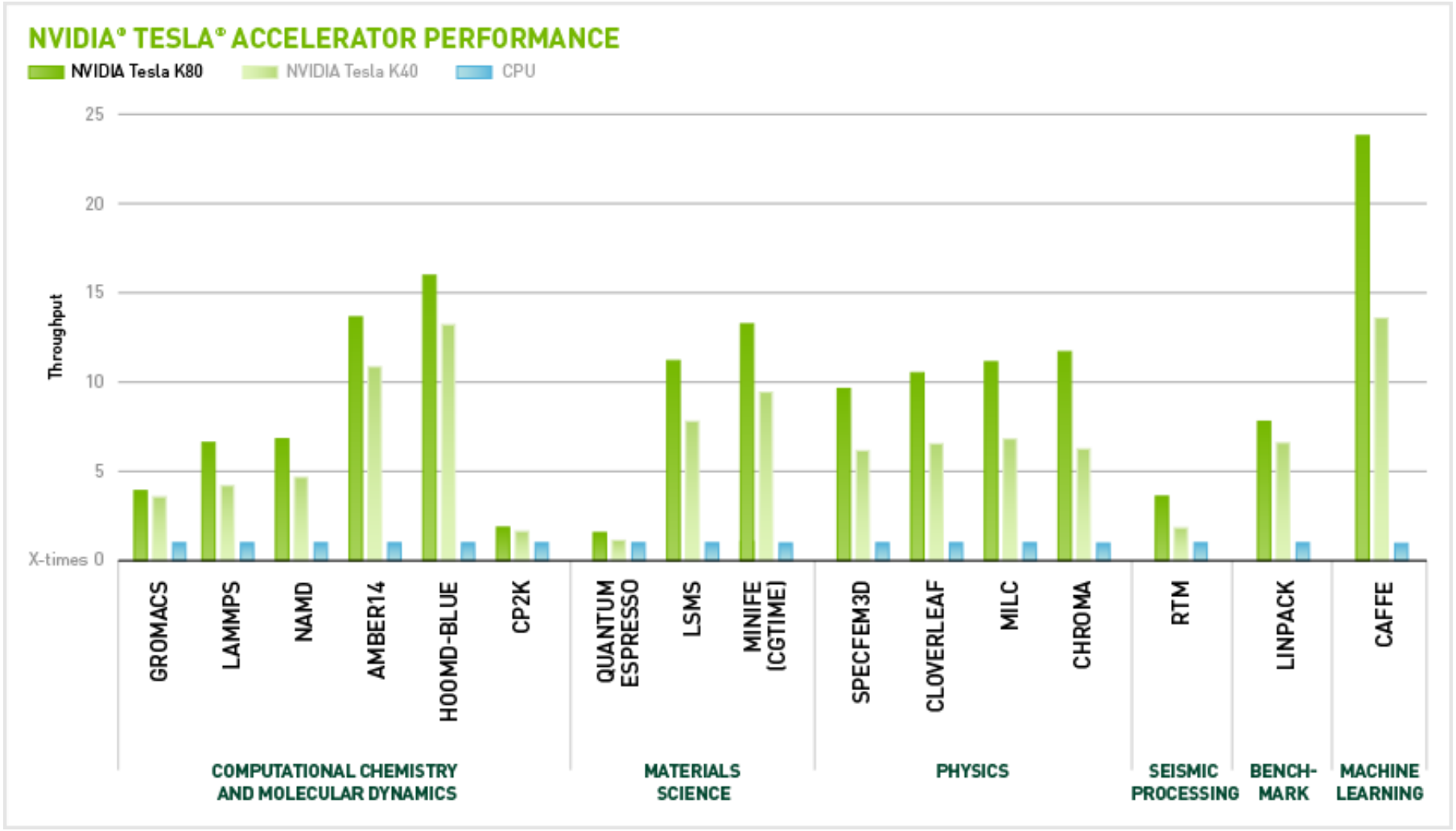


	1/12	2/12	3/12	4/12	1/13	2/13	3/13	4/13	1/14	2/14	3/14	4/14	1/15	2/15	3/15	4/15	1/16	2/16
AMD	38%	40%	36%	34%	36%	38%	36%	35%	35%	38%	28%	24%	23%	18%	19%	21%	23%	30%
nVidia	62%	59%	64%	66%	64%	62%	64%	65%	65%	62%	72%	76%	77%	82%	81%	79%	77%	70%

Опрос пользователей (2016-2017)

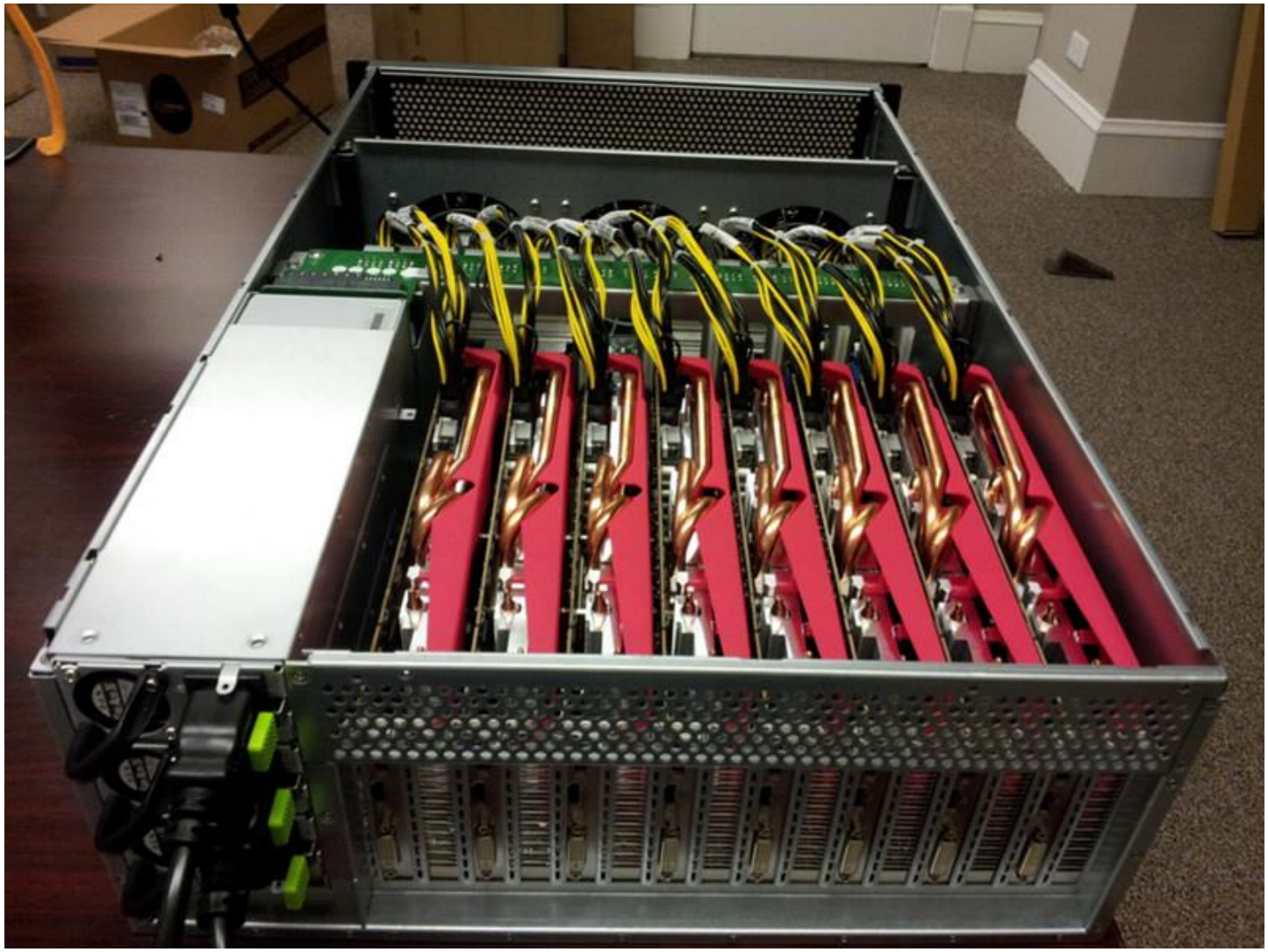


Причины использования GPU для вычислений

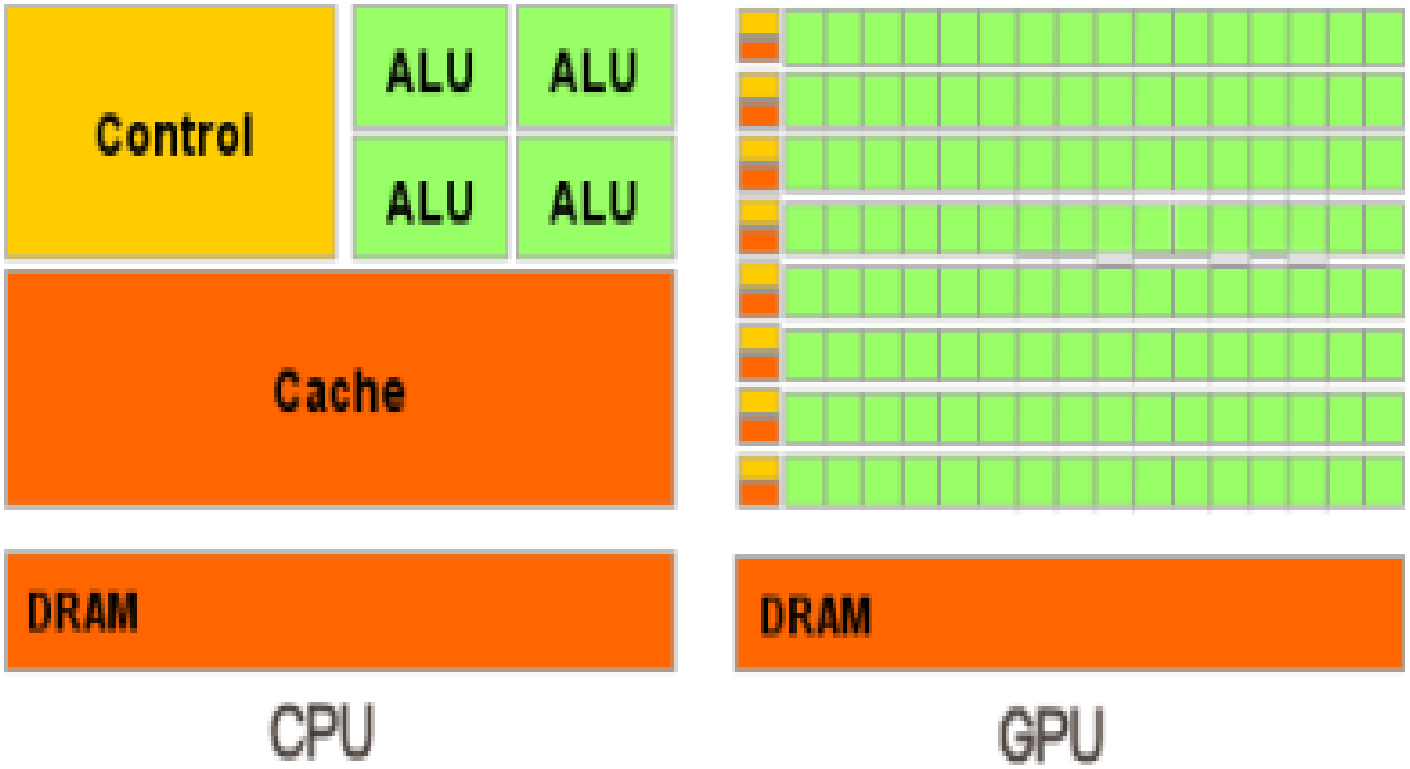


*С официального сайта компании NVIDIA

Пример серв. платформы с GPU



Причины использования GPU для вычислений

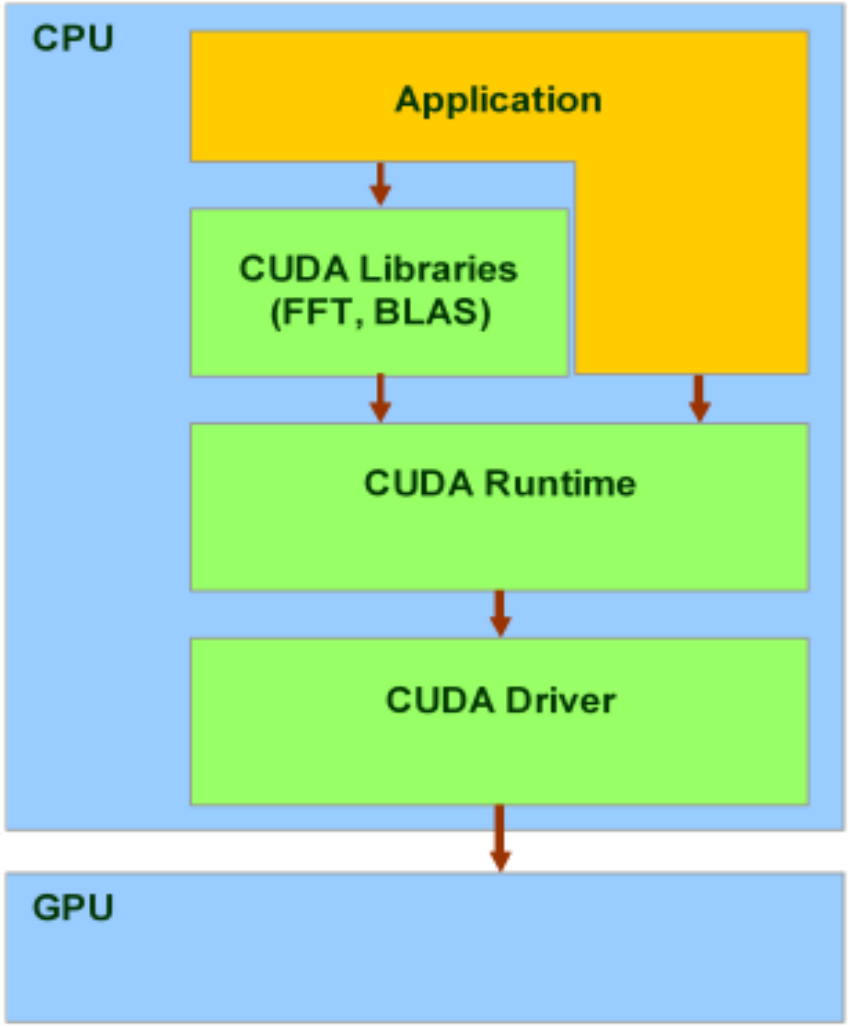


Инструменты для GPGPU

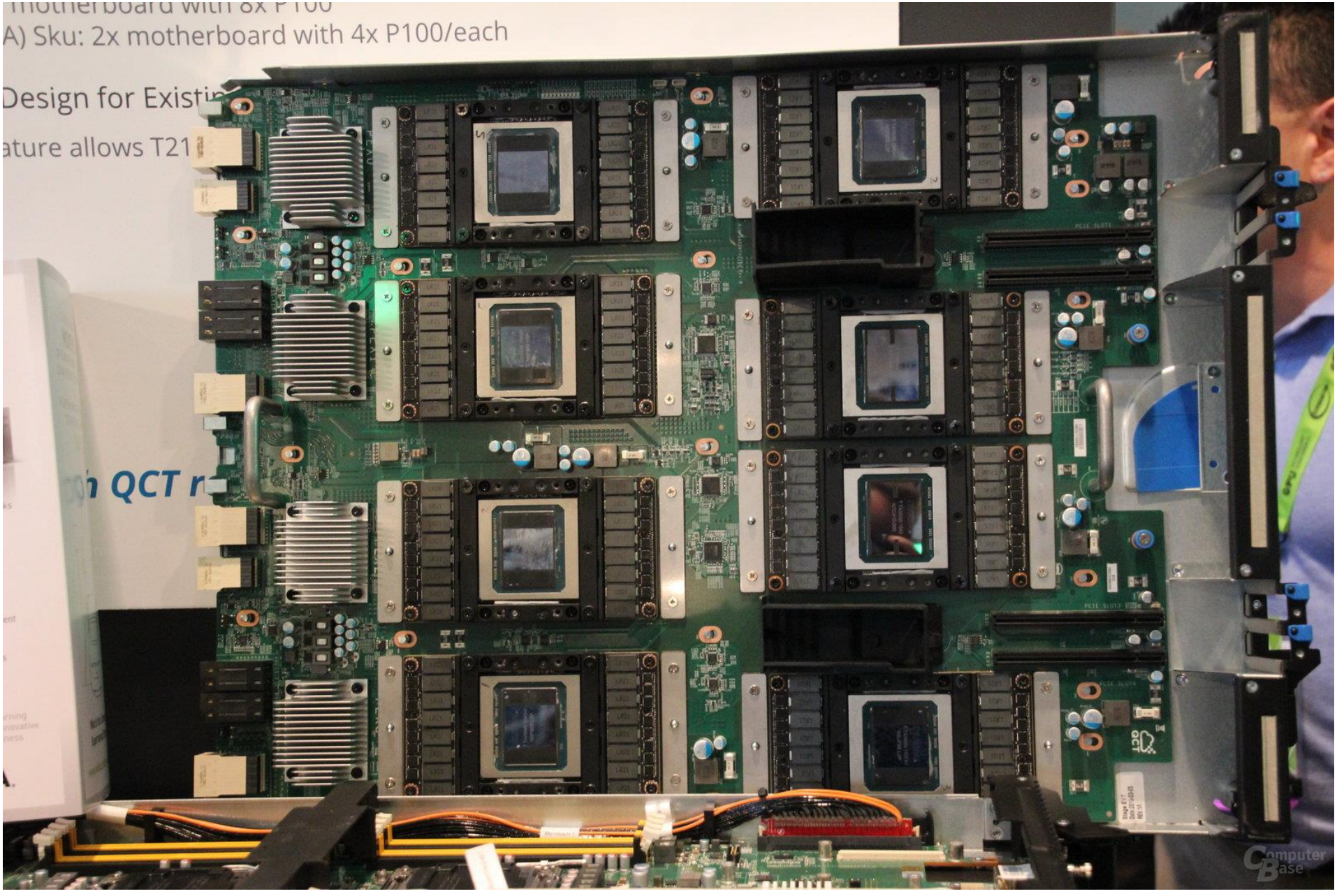
CUDA — хороший фреймворк для видеокарт NVIDIA (аппаратно-зависимая платформа)
Firestream — фреймворк для GPGPU вычислений на видеокартах AMD. (аппаратно-зависимая платформа)
HLSL — шейдерный язык DirectX (аппаратно-зависимая платформа)
DirectCompute — прикладной язык программирования (DirectX)
GLSL — шейдерный язык OpenGL

OpenCL — аппаратно и программно-независимая платформа для вычислений на CPU и GPU (нуждается в существенных доработках)

Состав CUDA

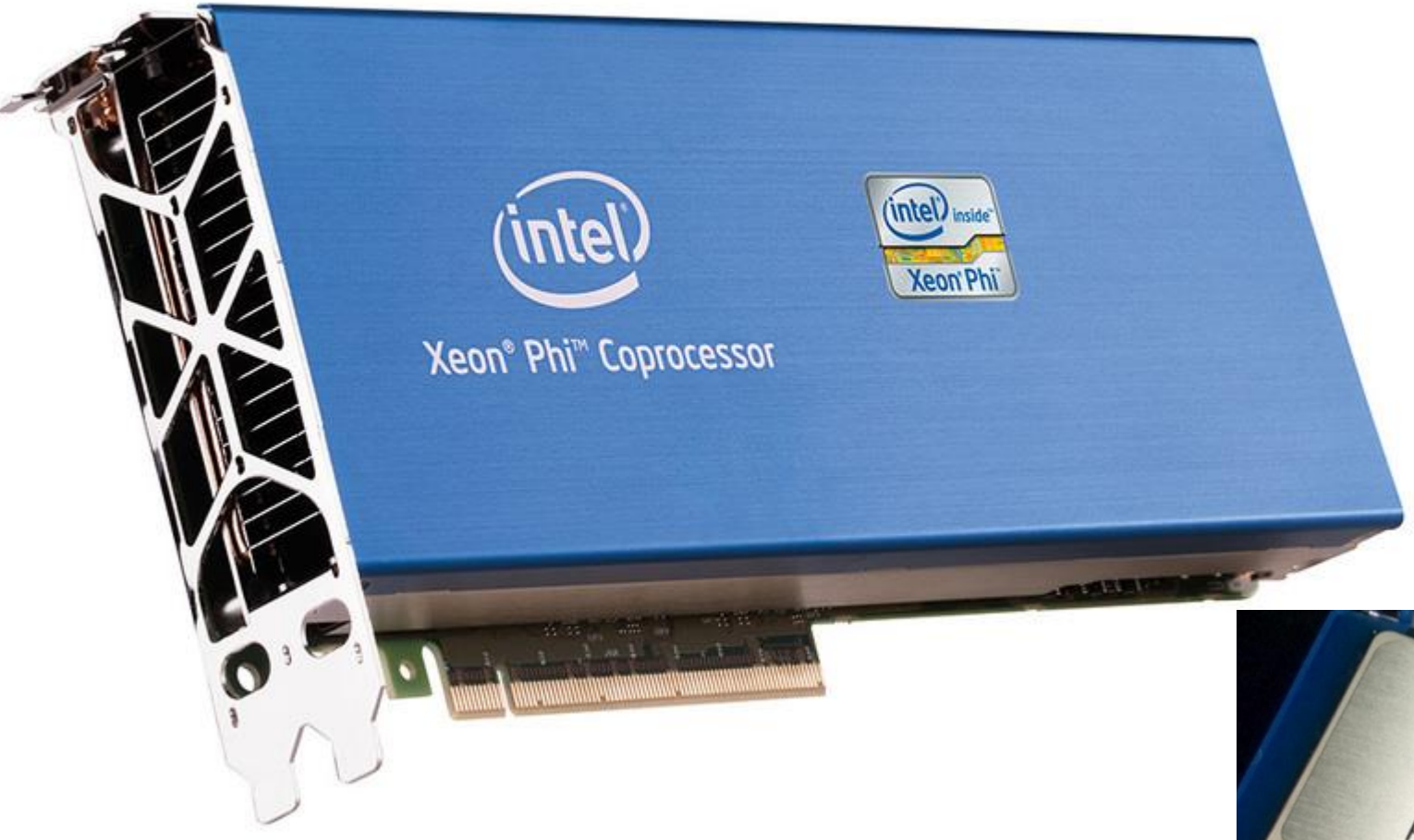

CPU
Application
CUDA Libraries
(FFT, BLAS)
CUDA Runtime
CUDA Driver
GPU

Изменение форм-фактора


A) Sku: 2x motherboard with 4x P100/each
Design for Existi
ature allows T21
h QCT r

Intel Xeon Phi

Сопроцессор Intel Xeon Phi



Архитектура MIC

- **Intel MIC** (*Many Integrated Core Architecture*) — архитектура многоядерной процессорной системы, разработанная Intel

В основе архитектуры MIC лежит классическая архитектура x86, на ускорителе исполняется ОС Linux.
Для программирования MIC предполагается использовать OpenMP, OpenCL, спец. компиляторы Intel Fortran, Intel C++.

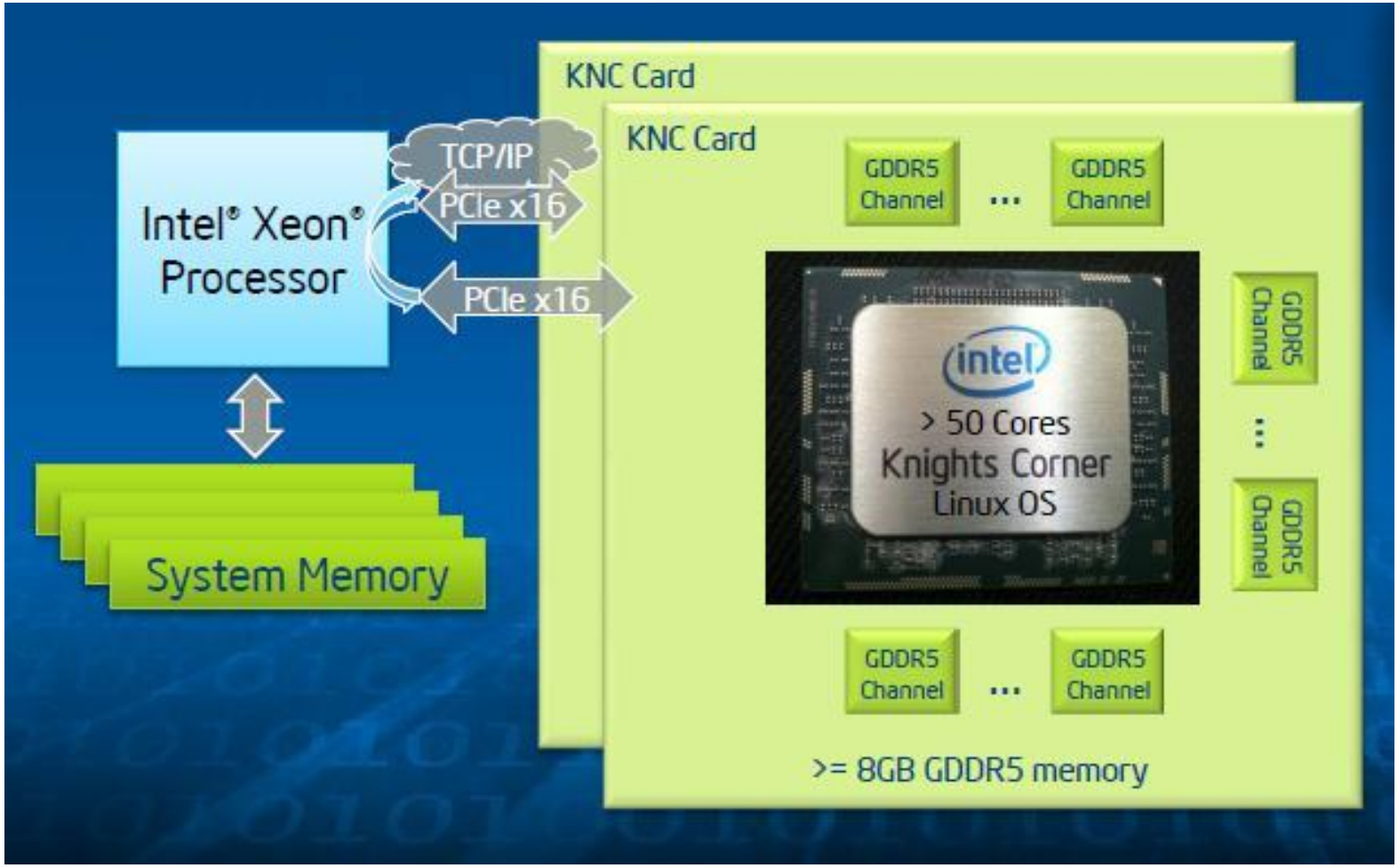
От предыдущих разработок унаследованы:

- набор команд x86;
- 512-битные векторные АЛУ;
- когерентный L2 кеш размером 512 КБ на ядро;
- кольцевая шина для связи ядер и контроллера памяти.

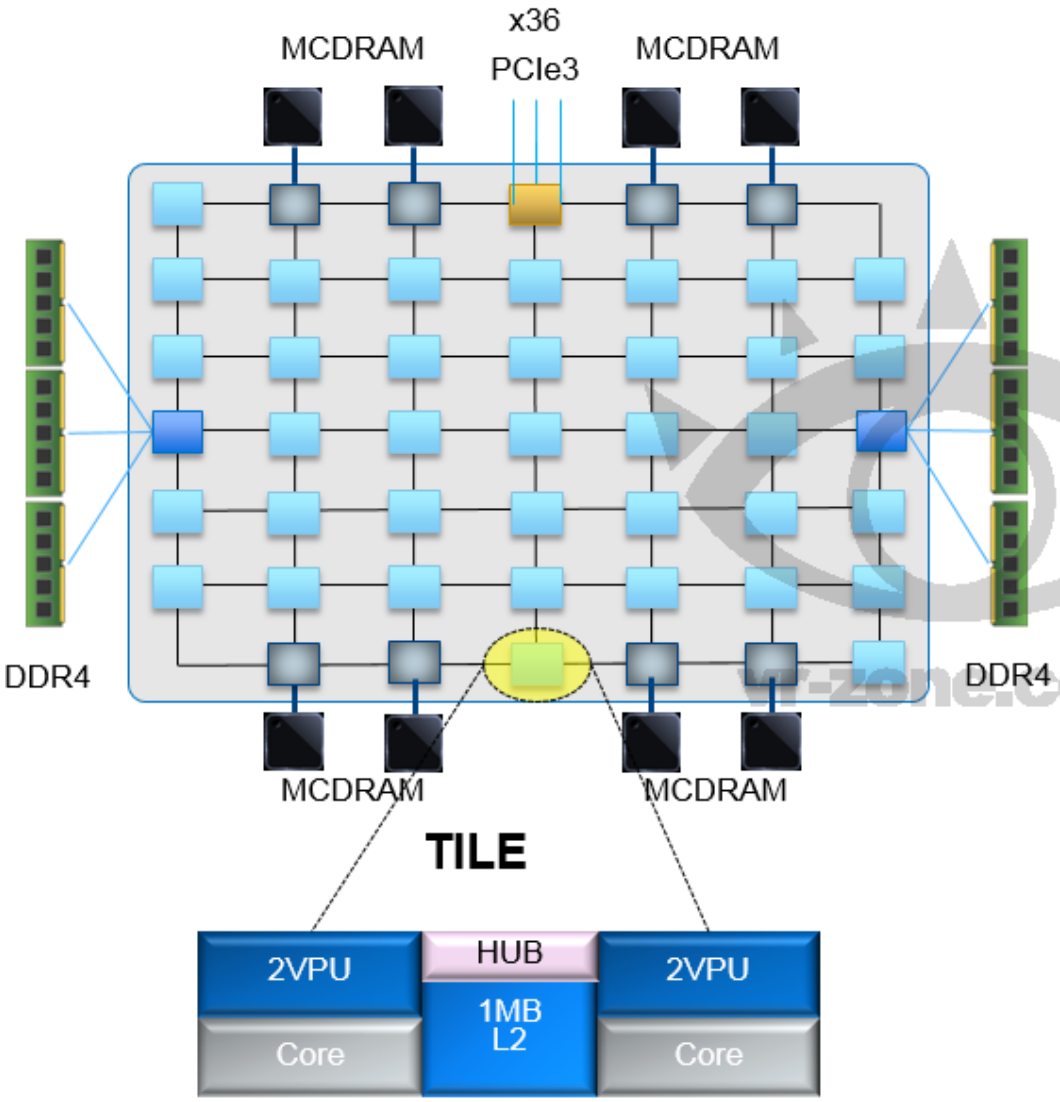
Поколения Intel Xeon Phi

1. Knights Ferry (2010) 45 нм
2. Knights Corner (2012) 22 нм
3. Knights Landing (2015) 14 нм
4. ~~Knights Hill (2017) 10 нм~~
5. Knights Mill (2017) 14 нм

Взаимодействие с процессором и памятью



Knights Landing Processor Architecture



Up to 72 Intel Architecture cores based on Silvermont (Intel® Atom processor)

- Four threads/core
- Two 512b vector units/core
- Up to 3x single thread performance improvement over KNC generation

Full Intel® Xeon processor ISA compatibility through AVX-512 (except TSX)

6 channels of DDR4 2400 MHz -up to 384GB

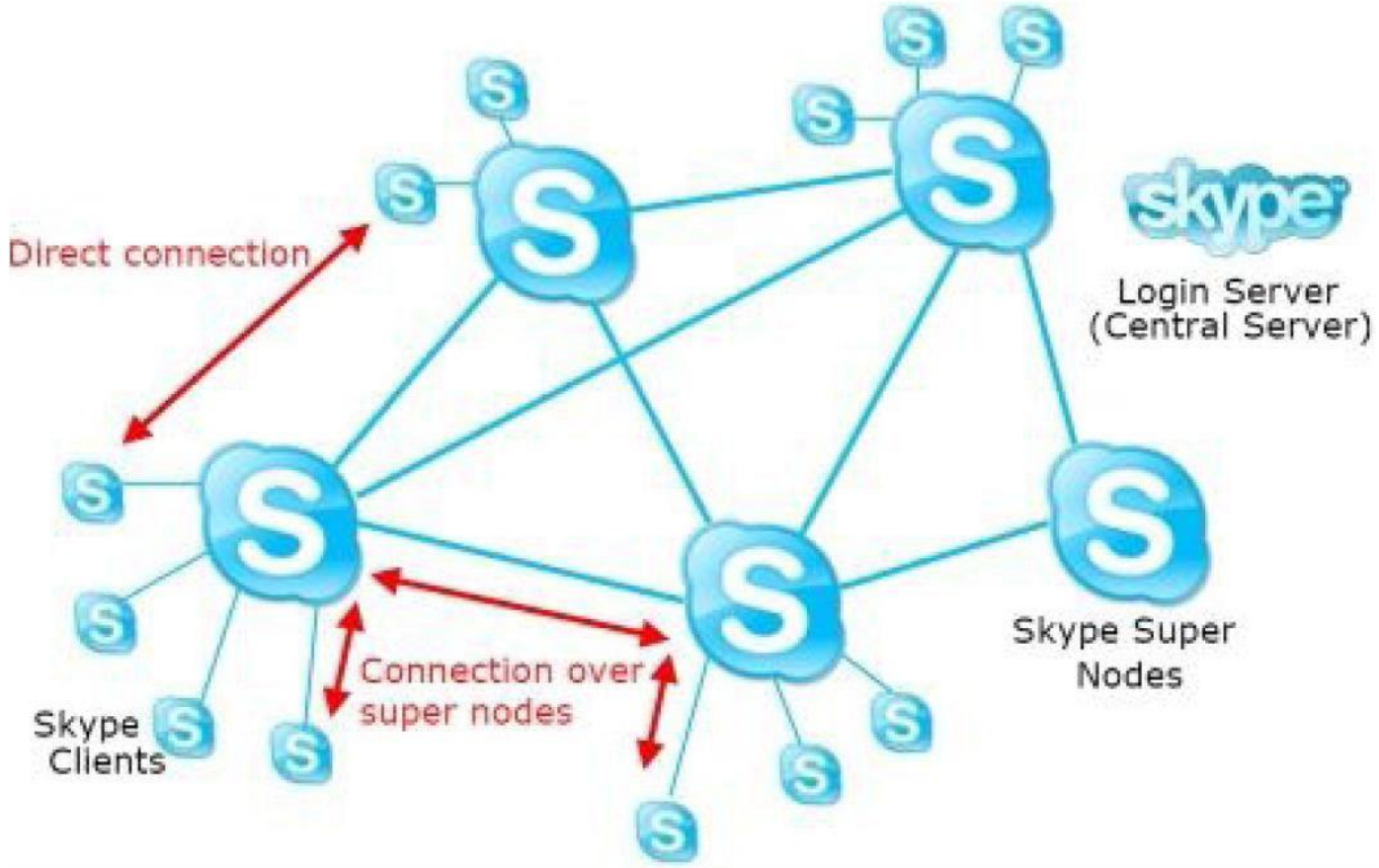
36 lanes PCI Express* Gen 3

8/16GB of high-bandwidth on-package MCDRAM memory >500GB/sec

200W TDP

Пиринговые сети и грид-системы

Пиринговые сети



Особенности грид-систем

- Гетерогенность узлов распределенной системы;
- Автономность расчетов на различных узлах и невозможность постоянной координации расчетов между узлами;
- Ненадежность связей и возможное отключение вычислительных узлов;
- Непостоянное время непрерывной работы узла и трудность расчета длительных заданий;
- Наличие ошибок и задержек при расчетах.

Тип задач «bag of tasks»

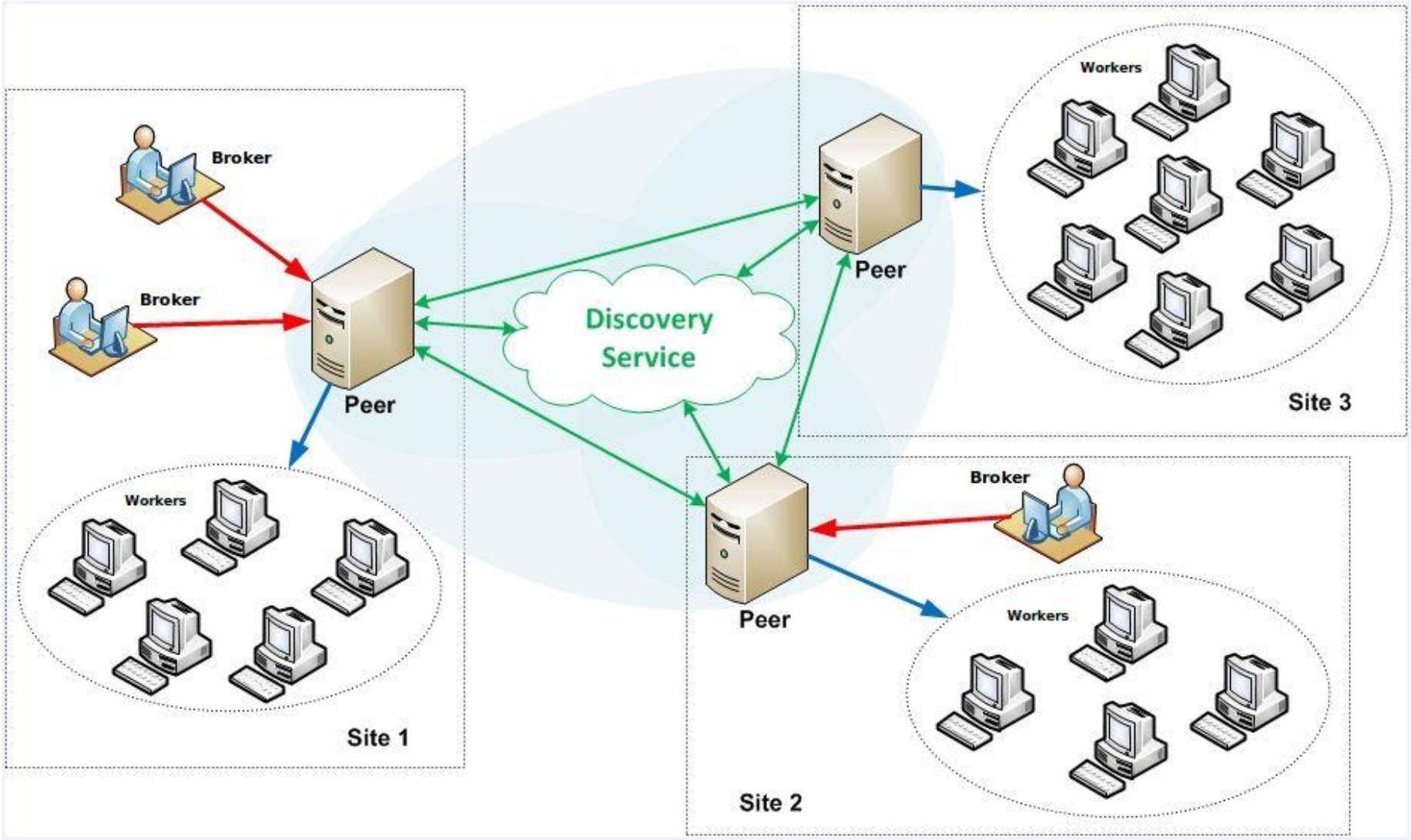
Задача может разбиваться на множество независимых подзадач. Каждая подзадача будет рассчитываться на отдельном вычислительном узле распределенной системы.

Для различных подзадач используются различные наборы входных данных и единый алгоритм их обработки. Такой тип задач в литературе называется «bag of tasks» или задача, разделяемая по данным.

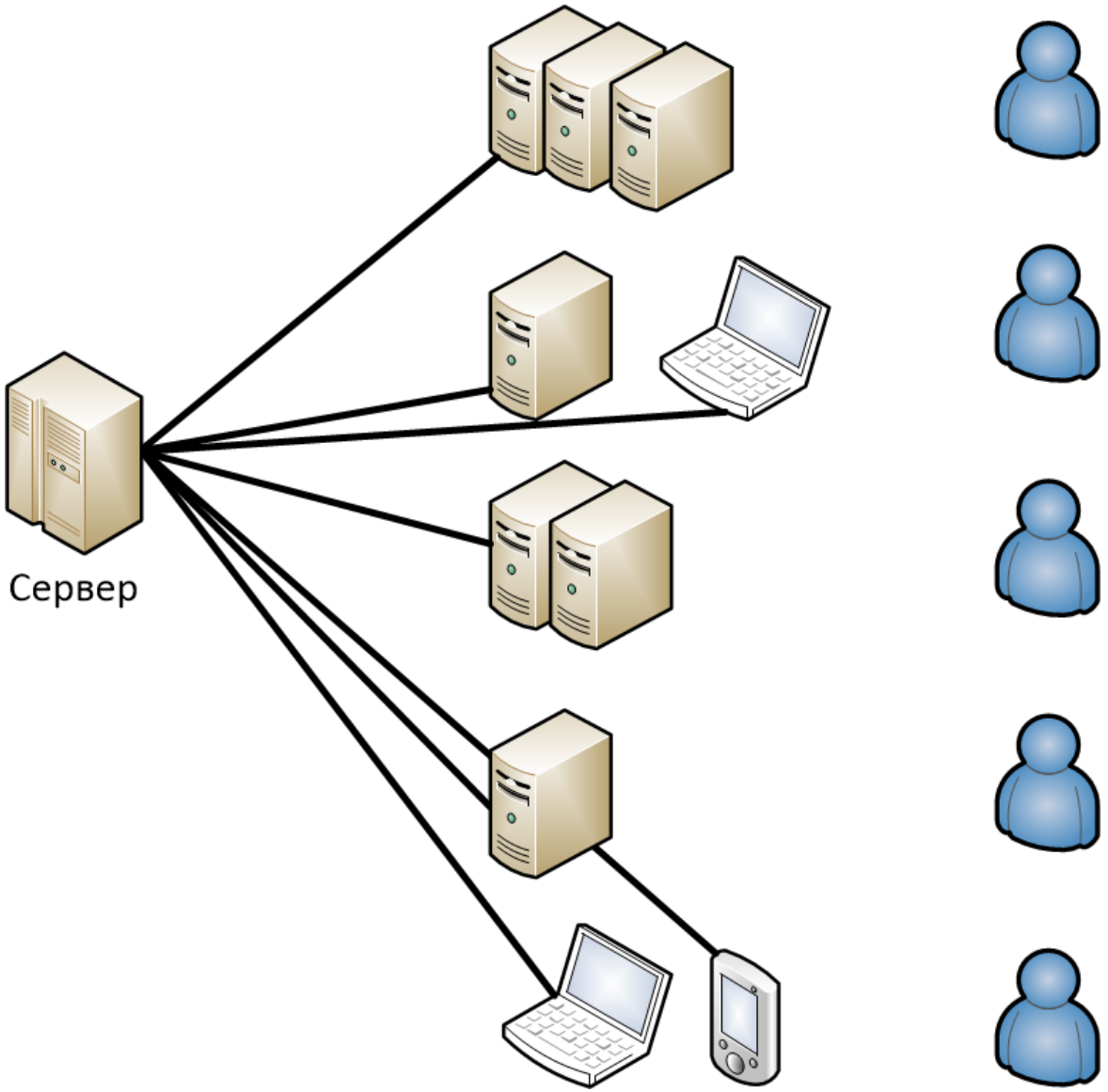
В качестве примера таких задач можно привести задачи:

- Задачи комбинаторики и полного перебора;
- SAT-задачи;
- задачи машинного обучения,
- задачи имитационного математического моделирования
- и др.

Грид-система OurGrid



Грид-система из персональных компьютеров



Платформы для организации распределенных вычислений

- HTCondor

- Globus

- BOINC

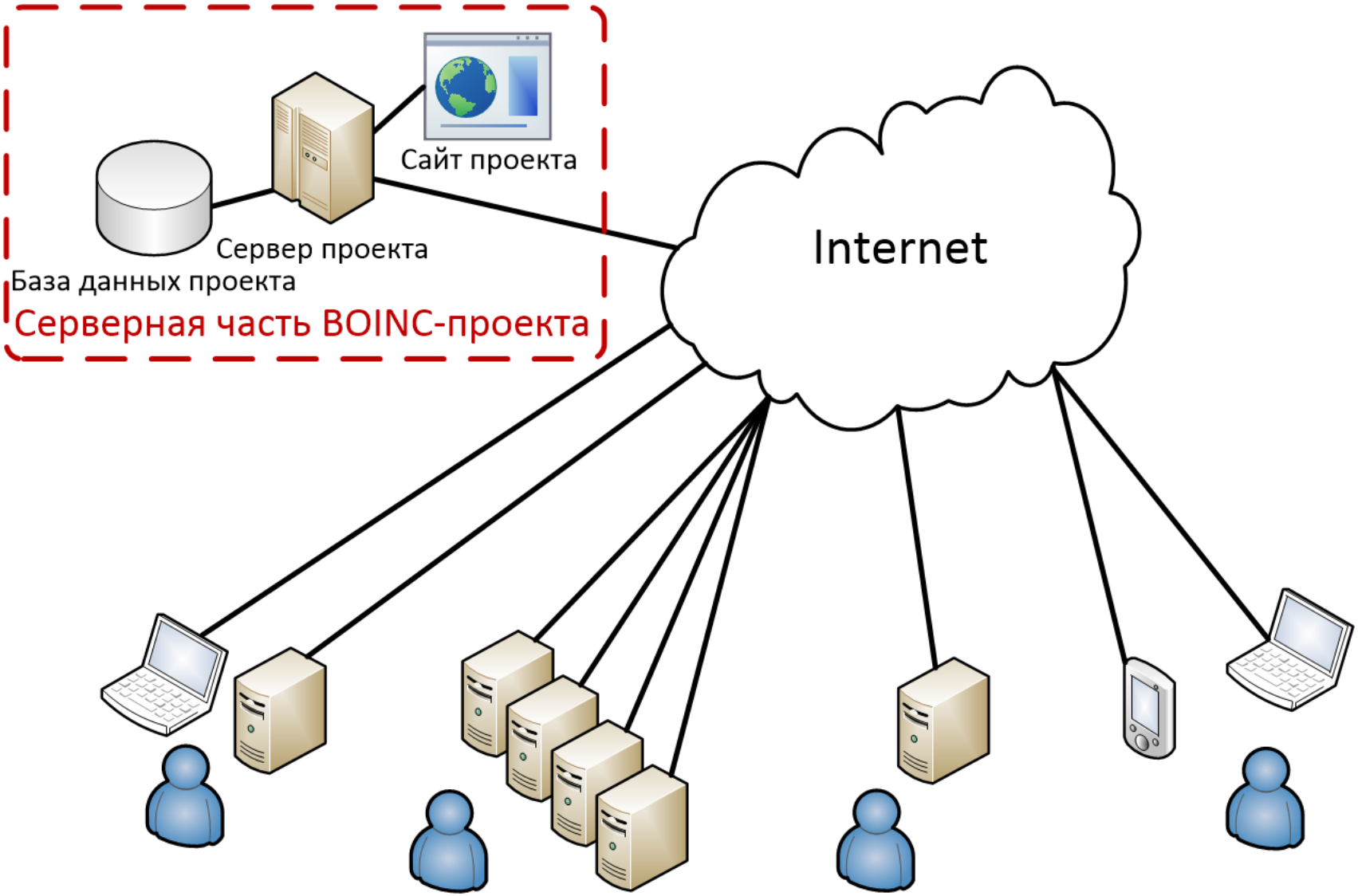
Платформа BOINC

BOINC – Berkeley Open Infrastructure for Network Computing

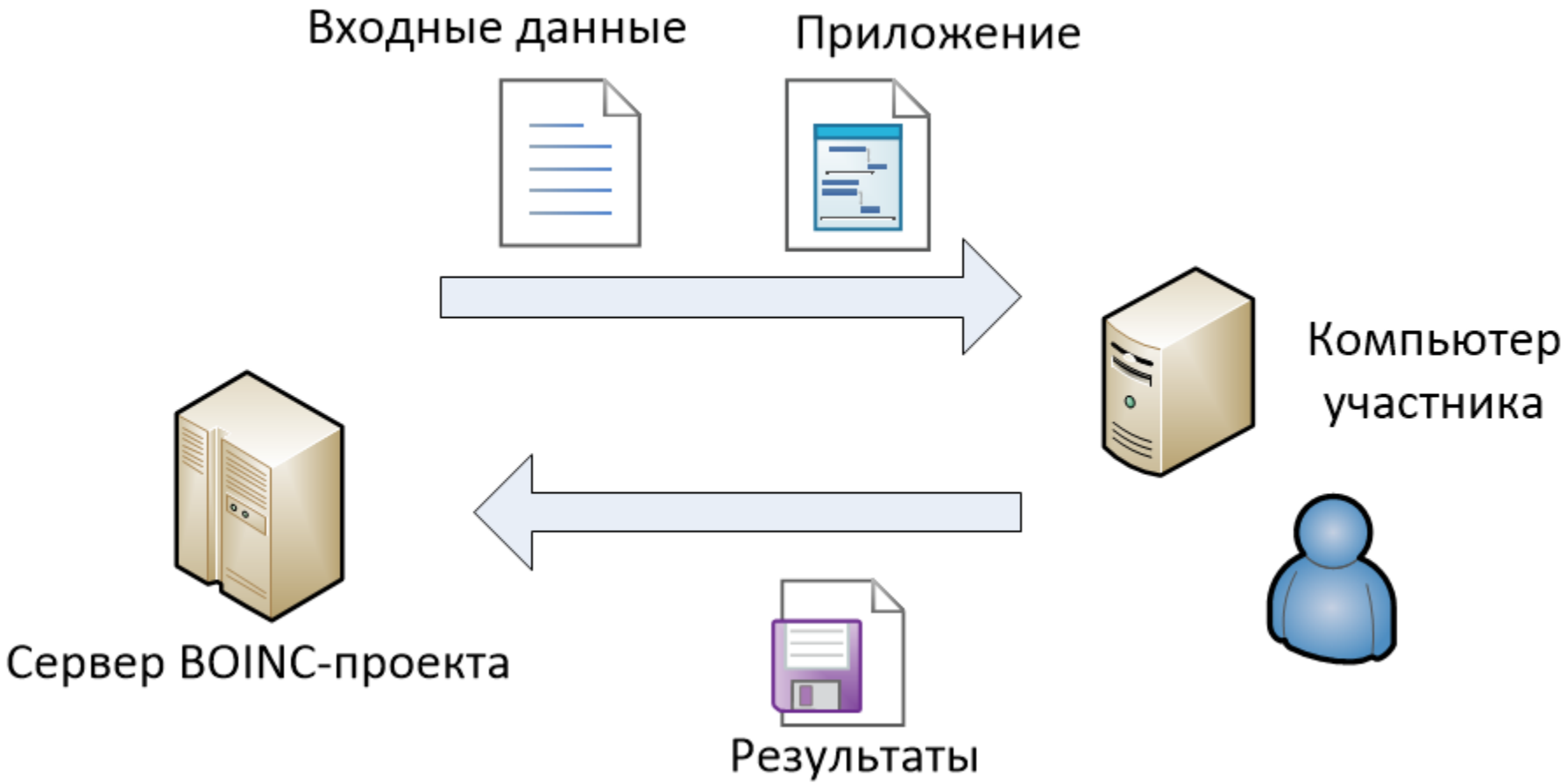
Платформа для организации распределенных вычислений:

- состоит из серверной и клиентской части;
- дает возможность задействовать вычислительные мощности персональных компьютеров(ПК);
- Кросплатформенная клиентская часть;
- Гибкая настройка клиентской части для эффективного использования ресурсов ПК.

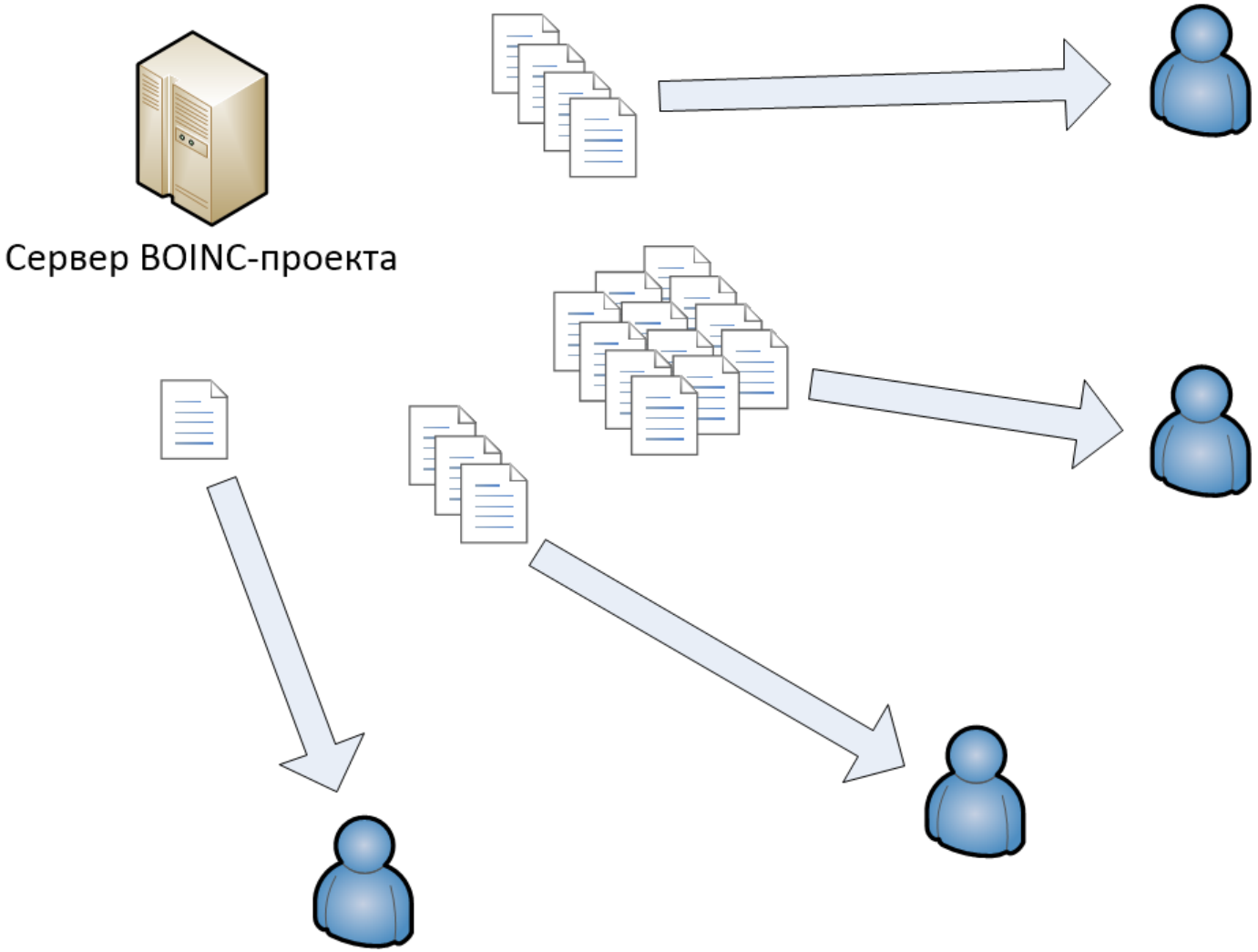
Схема подключения пользователей к проекту



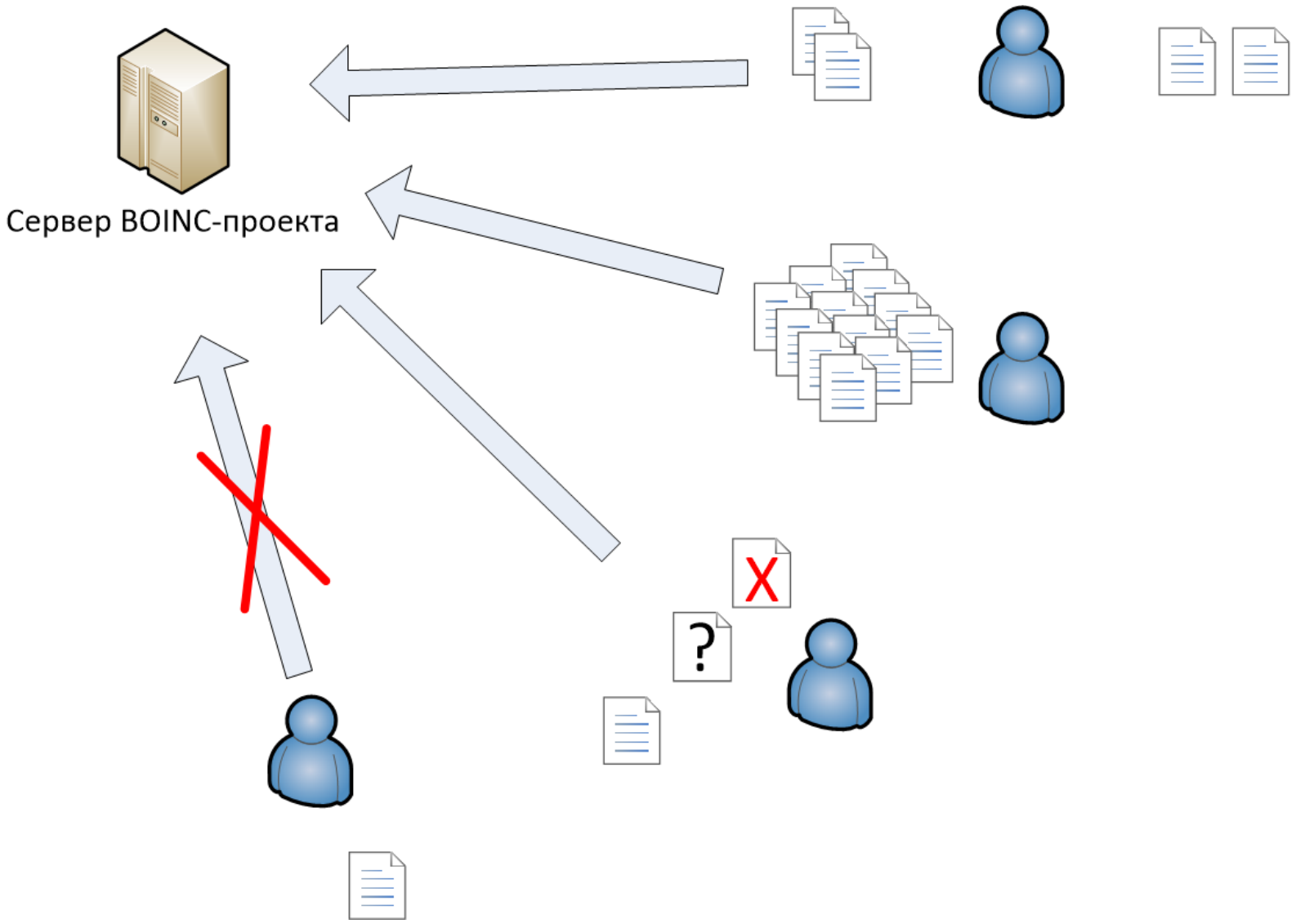
Обмен данными



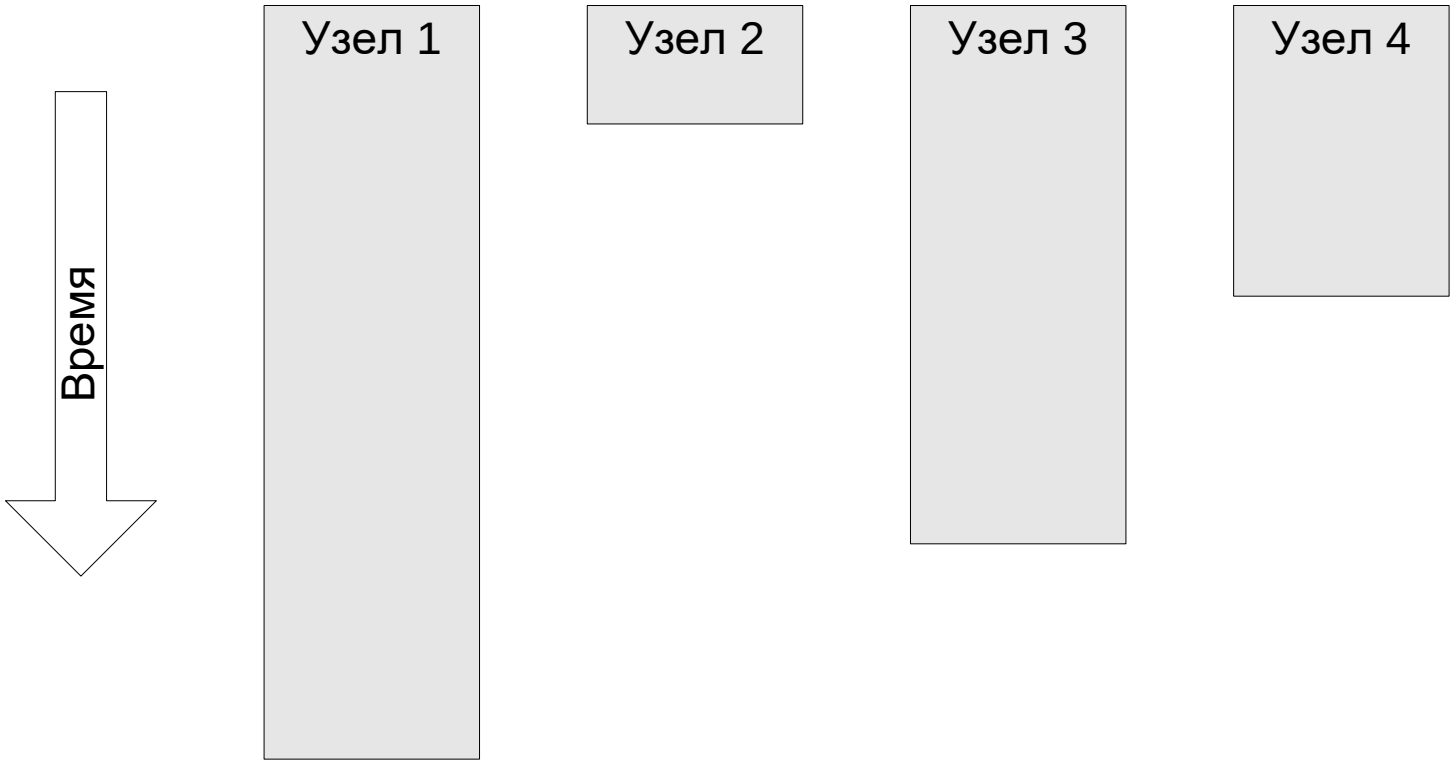
Выдача заданий участникам проекта



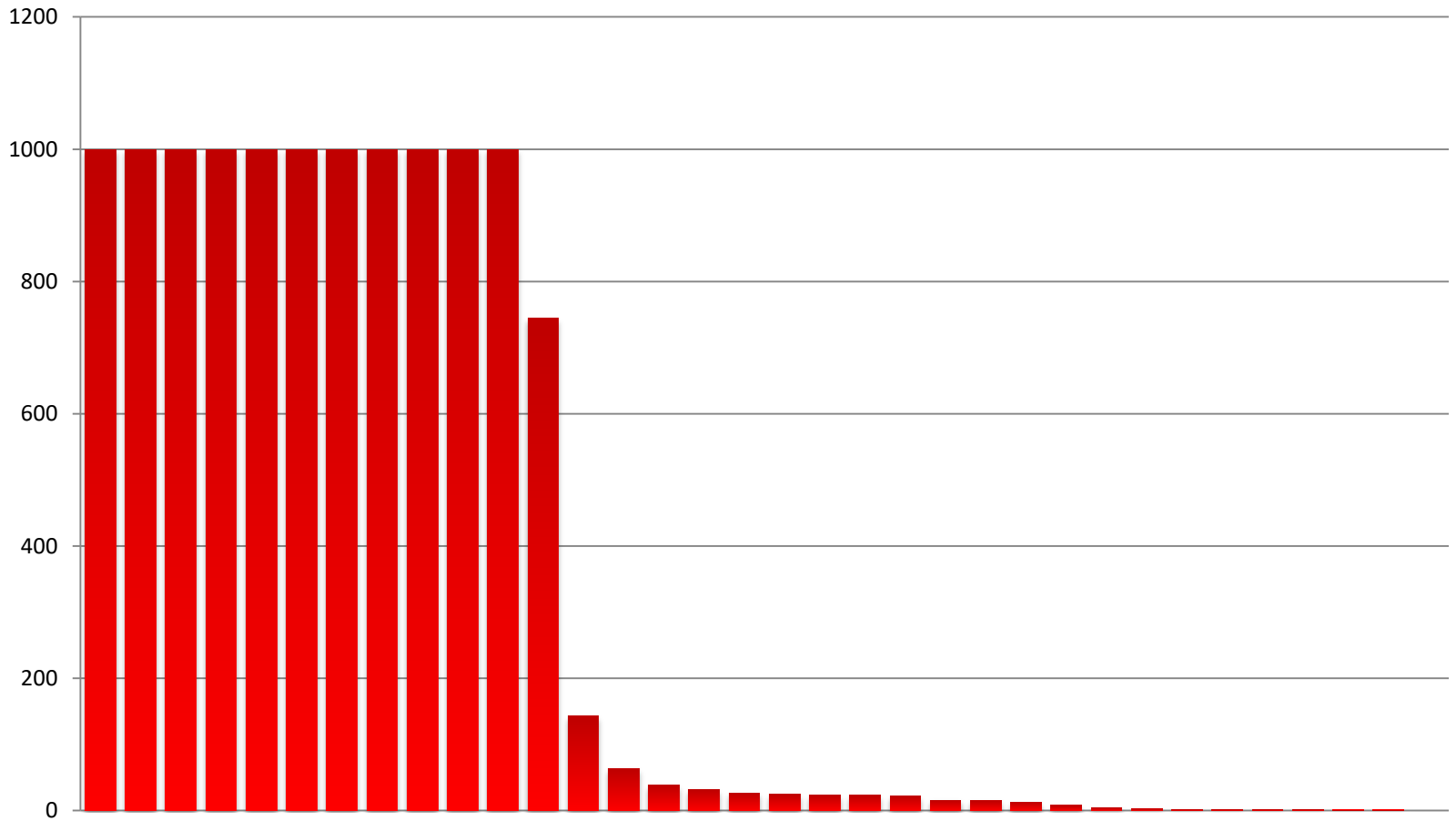
Получение результатов



Пример причины потери эффективности при распределенной реализации

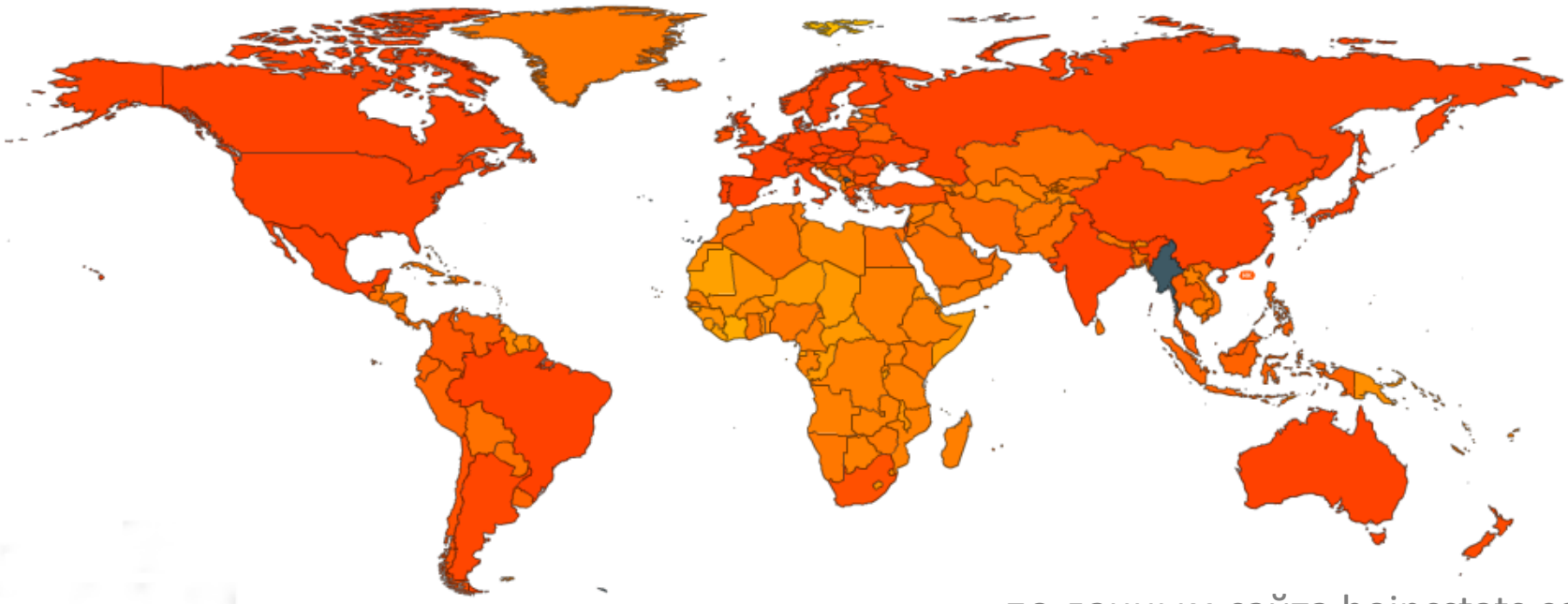


Образование «хвостов»



Добровольные распределенные вычисления (ДРВ) на платформе BOINC

- 4.3 миллиона участников
- 16 миллионов компьютеров
- Около 100 международных проектов добровольных распределенных вычислений



по данным сайта boincstats.com

Российские проекты ДРВ

- **SAT@home** (криптография, SAT-подход)
- **Optima@home** (решение задач конечномерной оптимизации) (зонтичный проект)
- **NetMax@home** (имитационное моделирование телекоммуникационных сетей)
- **USPEX@home** (поиск новых материалов)
- **Acoustics@home** (исследование дна Японского моря)
- **AndersonAttack@home** (криптография)
- Gerasim@home (комбинаторика, латинские квадраты)
- XANSONS for COD (материаловедение)
- ODLK@home (поиск канонических форм латинских квадратов)
- RakeSearch (поиск пар латинских квадратов)
- Amicable Numbers (поиск дружественных чисел)

Особенности организации экспериментов в проекте ДРВ

- Необходимость взаимодействия с добровольцами
- Общение на форуме
- Научно-популярное объяснение задач проекта
- Своевременное обновление информации на сайте проекта
- Обеспечение достаточного количества заданий для постоянной работы проекта
- Большой объем результатов и входных данных
- Необходимость обработки полной серии результатов
- Минимизация вероятности ошибочного расчета подзадачи
- Необходимость начальной репликации заданий в связи ненадежностью узлов