

Московский государственный технический университет им. Н.Э.Баумана
31 октября - 4 декабря 2022



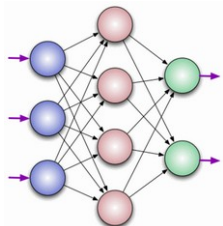
Студенческий практикум МГТУ им. Н.Э.Баумана по обработке и визуализации графов

Алексей Юрьевич Попов, Станислав Вадимович Ибрагимов,
Егор Николаевич Дубровин, Ли ЦзяЦзянь, Михаил Гейне

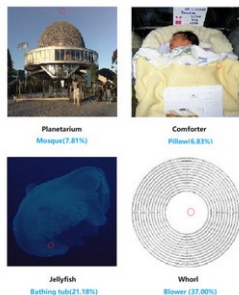
Московский государственный технический университет
им. Н.Э. Баумана, Москва, РФ

Проблемы слабого искусственного интеллекта

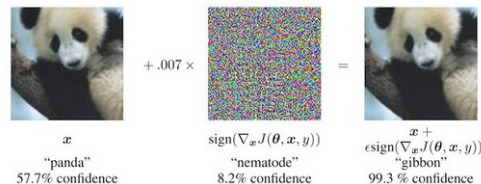
Искусственные нейронные сети генерируют выходные данные для любого входного шаблона



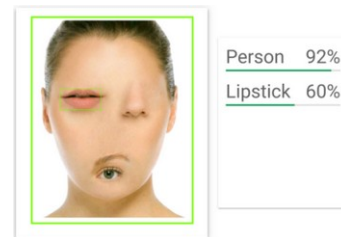
Результат обучения нейронной сети не всегда предсказуем



Внесение шума в изображение существенно снижает точность распознавания



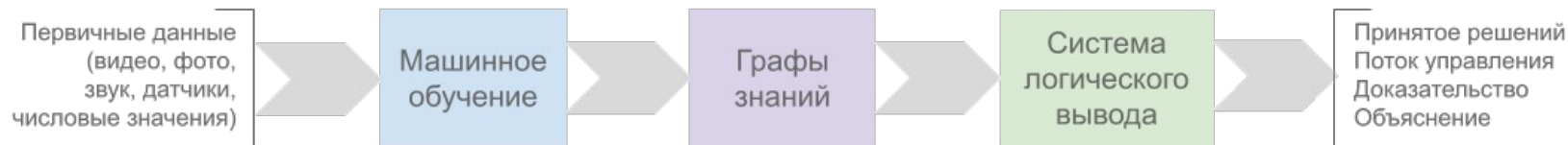
Трансляционная инвариантность приводит к ошибкам



Дэниэлл Денетт,
философ из Университета
Тафтса

«Я считаю, что если мы собираемся использовать эти вещи и зависеть от них, тогда нужно понимать, как и почему они действуют так, а не иначе. Если они не могут лучше нас объяснить, что они делают, то не стоит им доверять».

Аналитическая система на основе графов знаний



Этого достигли ИТ

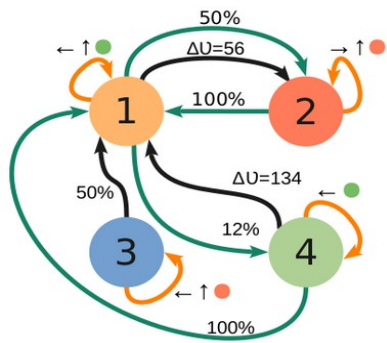
- Структуры/способы/методы/алгоритмы статичны.
- Структура вычислителя определяется на основе принципов универсальности.
- Структура программных систем определяется применяемыми технологиями.
- Информация в большинстве случаев представляется в виде реляционных моделей.
- Технический эффект достигается в рабочем режиме информационной системы.

Это сделала природа

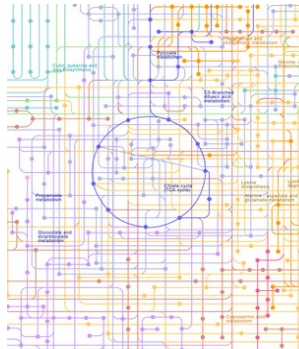
- Живой организм непрерывно обучается с момента рождения до смерти, при этом обеспечивает свою жизнедеятельность.
- Окружающая действительность определяют физиологические особенности, которые передаются последующим поколениям.
- Знания передаются различными способами.
- Человек проходит около 12 различных стадий на жизненном пути.

Представление знаний

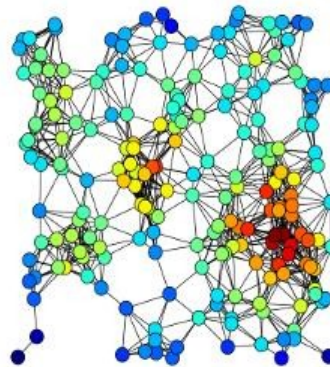
Знания представляются в виде графовых моделей, позволяющих однозначно интерпретировать результат. Вершины и ребра графа представления знаний обладают атрибутами, которые анализируются алгоритмами и позволяет делать логический вывод.



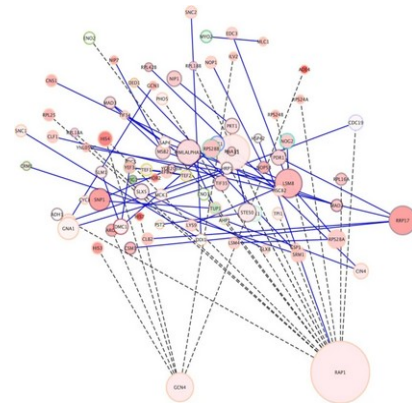
Динамический граф
сцены



Фрагмент графа
обмена веществ



Граф результатов
анализа контрагентов
для участника рынка



Граф белок-
белковых
взаимодействий

Применение графов в биологии и медицине

- Интерактомика
- Анализ проблемы индивидуальной нормы
- Моделирование и визуализация процессов в биологических системах
- Моделирование и анализ популяций и сложных сообществ



Существующие подходы к обработки графов

Режимы обработки

- Обработка статичных графов
- Поточковая обработка статичных графов
- Обработка динамических графов

Программные решения

Эффективные структуры данных
Библиотеки обработки графов
Графовые базы данных

Аппаратные решения

Многопоточность и многонитевость
Графические ускорители
Специализированная память
Ускорители на ПЛИС

Набор команд дискретной математики DISC

Discrete math operations	Description	DISC instructions
$A = \langle A_1, \dots, A_n \rangle$	- store function of n sets as an A tuple	Insert
$R(A_i, x, y), x \in A_i, y \in A_i$	- relationship between the x and y in the set A_i	Next/Previous/Neighbors
$ A_i , i = 1, n$	- cardinality of the A_i set	Cardinality
$x \in A_i, x \notin A_i, i = 1, n$	- check the inclusion/exclusion of the x in the set	Search
$A_i \cup x, i = 1, n$	- inserting the x into the set	Insert
$A_i \setminus x, i = 1, n$	- removing an element x from the set	Delete, Delete structure
$A \setminus A_i$	- removing the set A_i from the tuple A	Delete structure
$A_i \subset A_j$	- inclusion relation of the set A_i in A_j	Slices
$A_i \equiv A_j$	- equivalence relation operation	Slices
$A_i \cup A_j$	- union operation of two sets	OR
$A_i \cap A_j$	- intersection operation of two sets	AND
$A_i \setminus A_j$	- difference operation	NOT
$A_i \triangle A_j$	- symmetric difference	—
\bar{A}	- complement of the A_i	NOT
$A_i \times A_j$	- Cartesian product operation	—
2^{A_i}	- Boolean operation	—

Набор команд дискретной математики DISC

Операции, основанные на поиске

Поиск по ключу *SRCH*
Поиск минимального *MIN*
Поиск максимального *MAX*
Поиск следующего *NEXT*
Поиск предыдущего *PREV*
Ближайший больший *NGR*
Ближайший меньший *NSM*

Операции добавления/удаления

Вставка *INS*
Удаление *DEL*
Удаление множества *DELS*

Операции И-ИЛИ-НЕ

Объединение множеств *OR*
Пересечение множеств *AND*
Дополнение множеств *NOT*

Операции среза

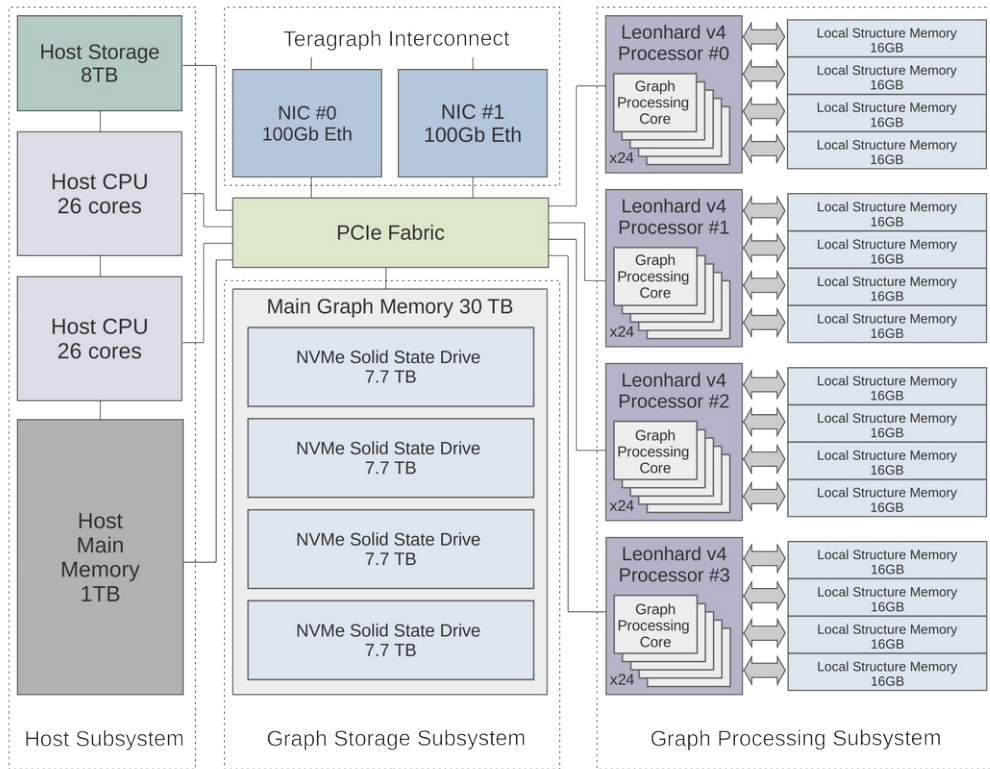
Срез больше *GR*
Срез больше или равно *GREQ*
Срез меньше *LS*
Срез меньше или равно *LSEQ*
Срез меньше/больше *GRLS*

Свойства множеств

Мощность множества *CNT*

Вычислительный узел комплекса Тераграф

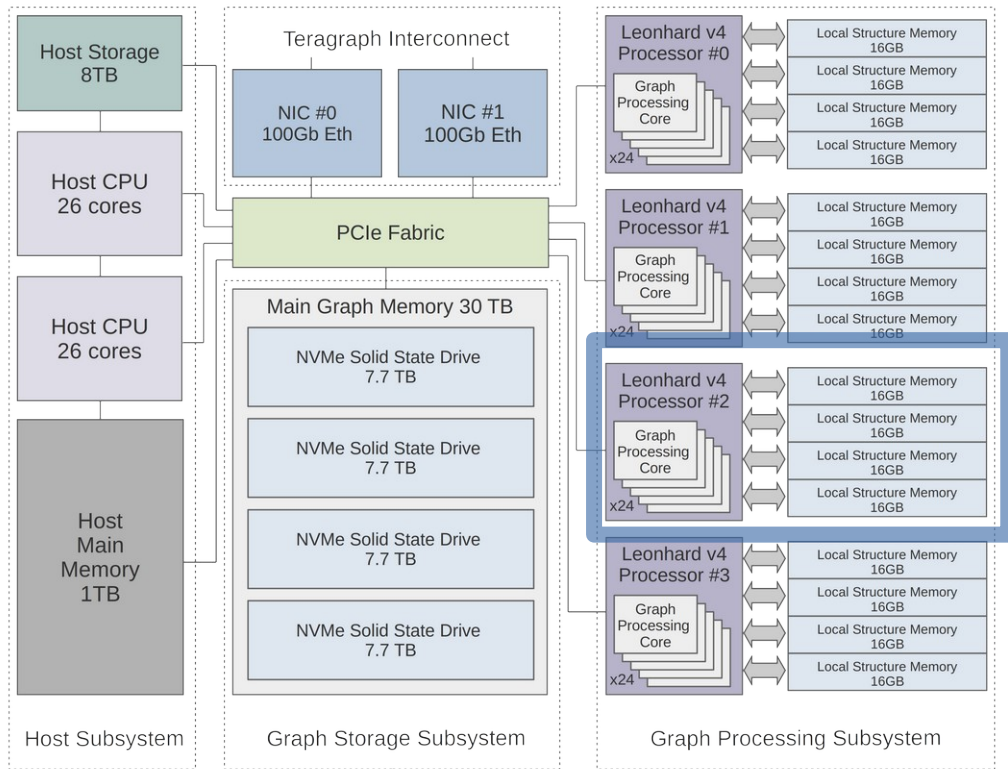
Teragraph node #1



- Предусмотрено длительное размещение графов в оперативном доступе.
- Используется ассоциативная память большого объема (2.5ГБ на одно ядро Graph Processing Core, GPC)
- Ядра GPC являются высокоэффективными гетерогенными системами, взаимодействующими через единой адресное пространство PCIe
- GPC самостоятельно обращается в локальное графовое хранилище 30ТБ и графовые хранилища других узлов
- Хост система выполняют второстепенные функции (инициализация, распределение и т.д.)

Вычислительный узел комплекса Тераграф

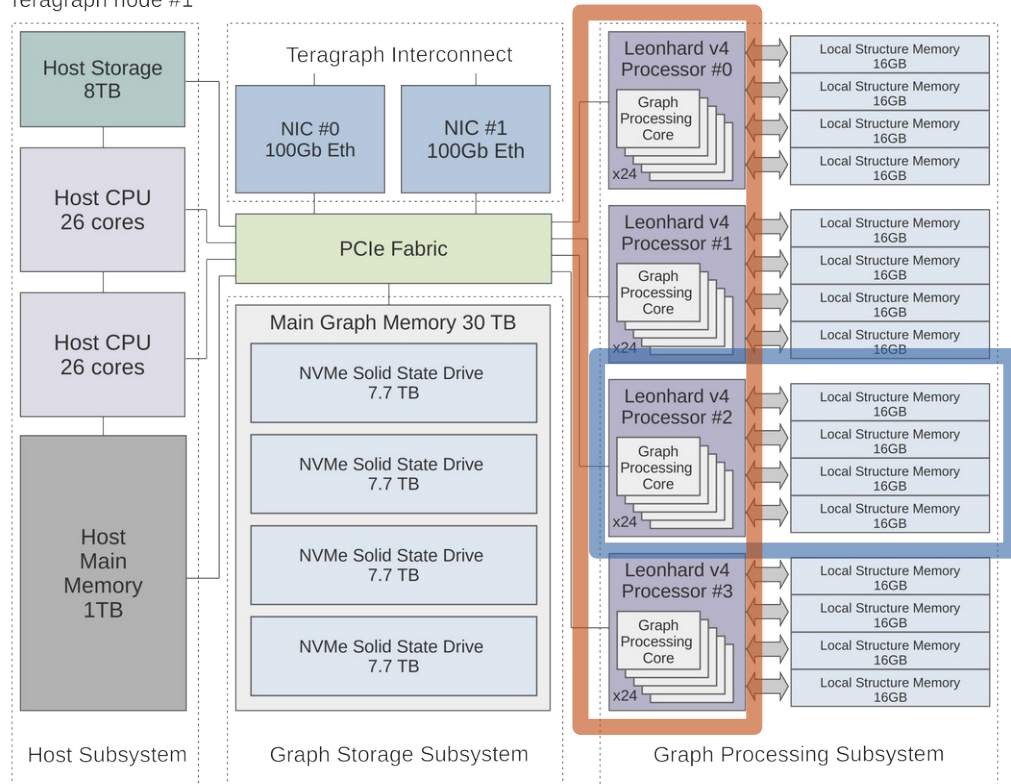
Teragraph node #1



- Предусмотрено длительное размещение графов в оперативном доступе.
- Используется ассоциативная память большого объема (2.5ГБ на одно ядро Graph Processing Core, GPC)
- Ядра GPC являются высокоэффективными гетерогенными системами, взаимодействующими через единой адресное пространство PCIe
- GPC самостоятельно обращается в локальное графовое хранилище 30ТБ и графовые хранилища других узлов
- Хост система выполняют второстепенные функции (инициализация, распределение и т.д.)

Вычислительный узел комплекса Тераграф

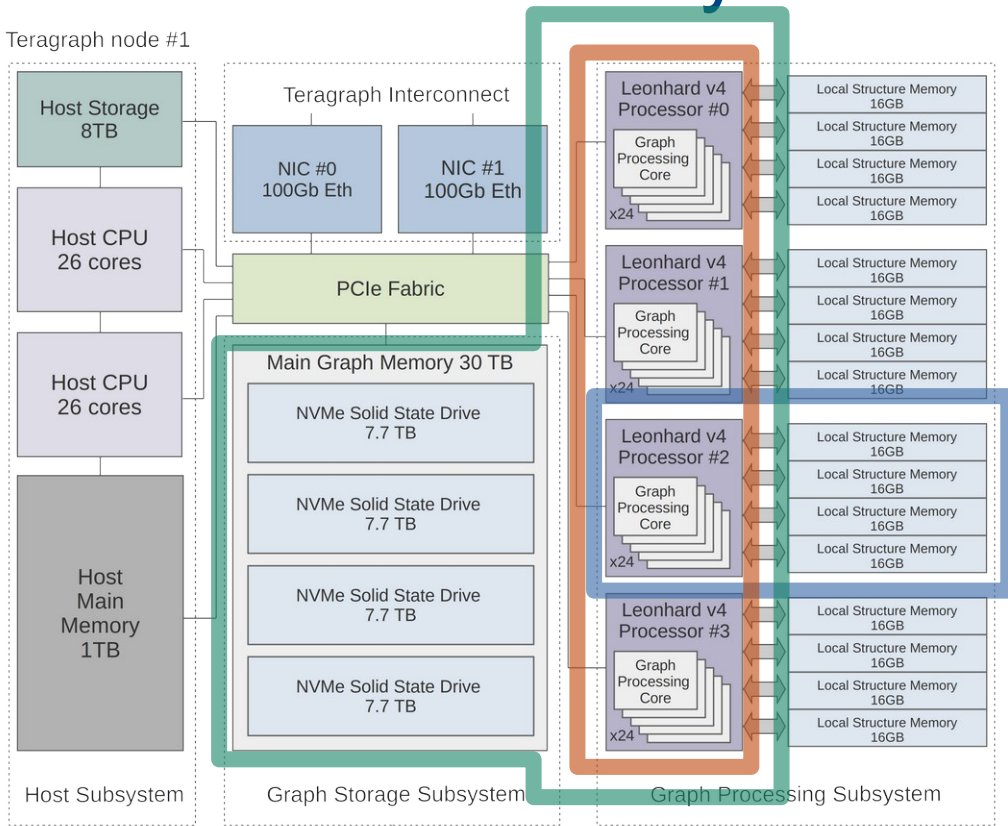
Teragraph node #1



- Предусмотрено длительное размещение графов в оперативном доступе.
- Используется ассоциативная память большого объема (2.5ГБ на одно ядро Graph Processing Core, GPC)
- Ядра GPC являются высокоэффективными гетерогенными системами, взаимодействующими через единой адресное пространство PCIe
- GPC самостоятельно обращается в локальное графовое хранилище 30ТБ и графовые хранилища других узлов
- Хост система выполняют второстепенные функции (инициализация, распределение и т.д.)

Вычислительный узел комплекса Тераграф

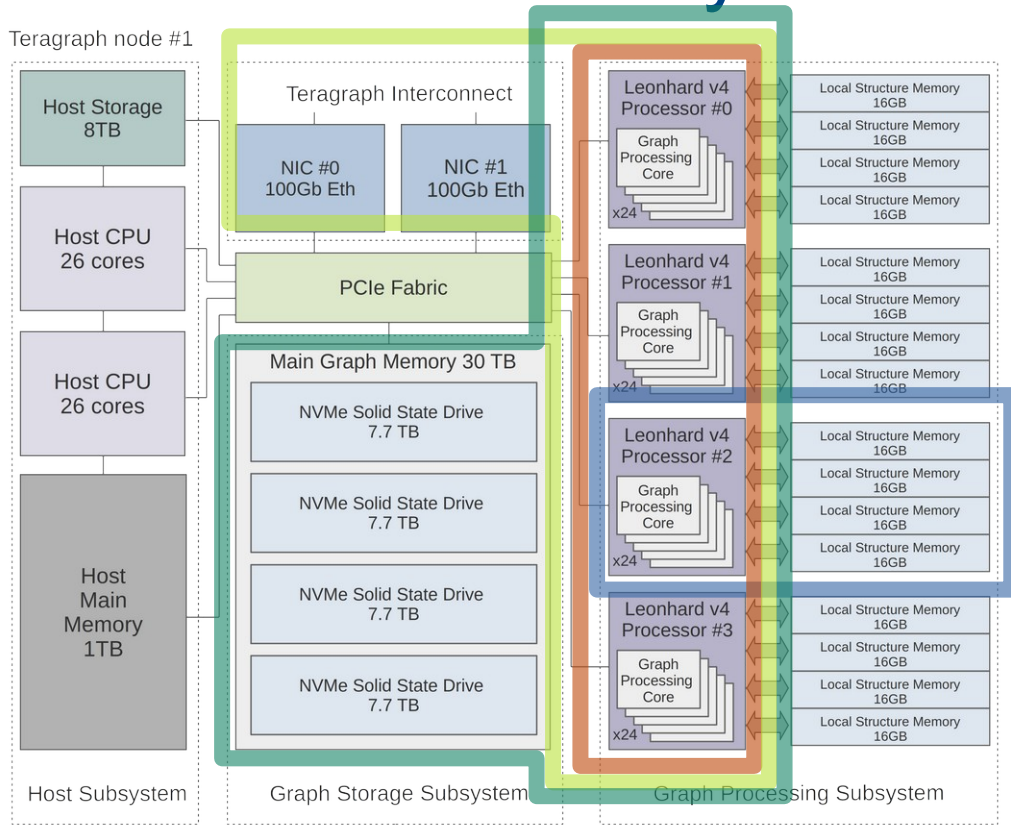
Teragraph node #1



- Предусмотрено длительное размещение графов в оперативном доступе.
- Используется ассоциативная память большого объема (2.5ГБ на одно ядро Graph Processing Core, GPC)
- Ядра GPC являются высокоэффективными гетерогенными системами, взаимодействующими через единой адресное пространство PCIe
- GPC самостоятельно обращается в локальное графовое хранилище 30ТБ и графовые хранилища других узлов
- Хост система выполняют второстепенные функции (инициализация, распределение и т.д.)

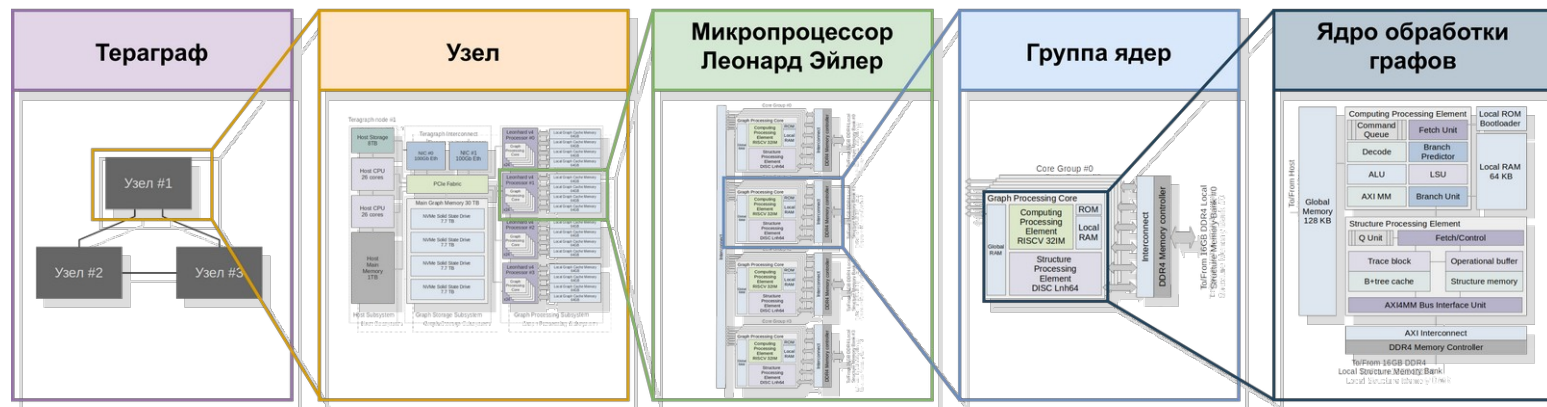
Вычислительный узел комплекса Тераграф

Teragraph node #1



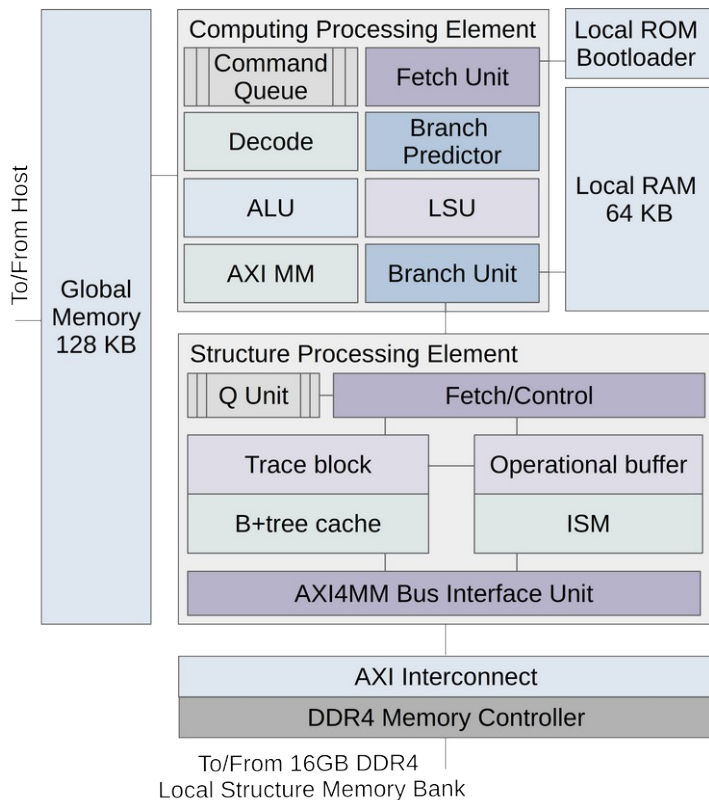
- Предусмотрено длительное размещение графов в оперативном доступе.
- Используется ассоциативная память большого объема (2.5ГБ на одно ядро Graph Processing Core, GPC)
- Ядра GPC являются высокоэффективными гетерогенными системами, взаимодействующими через единой адресное пространство PCIe
- GPC самостоятельно обращается в локальное графовое хранилище 30ТБ и графовые хранилища других узлов
- Хост система выполняют второстепенные функции (инициализация, распределение и т.д.)

Архитектура комплекса Тераграф



Характеристика	Значение
Количество процессоров Леонард Эйлер	9
Количество GPC	216
Кэш память (DDR4, ГБ)	576
Оперативная память GPC (ТБ)	48
Количество хранимых ключей	1 триллион

Гетерогенное ядро обработки графов

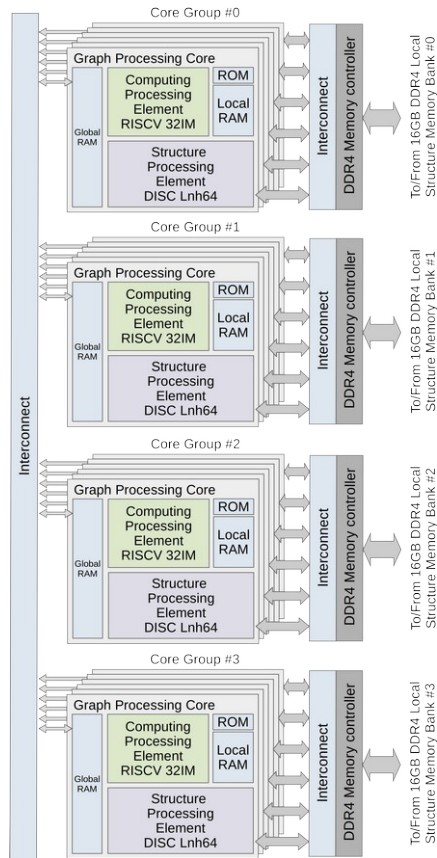


- GPC состоит из двух тесно связанных микропроцессоров: Computing Processor Element (CPE) и Structure Processing Element (SPE).
- CPE реализован на базе микропроцессора с набором команд `riscv32im`.
- SPE представляет собой микропроцессор с набором команд дискретной математики DISC
- SPE подключен, как ускорительное ядро к шине памяти CPE.
- Производительность GPC сопоставима с производительностью одного ядра Intel Xeon Platinum v8 при 10x меньшей частоте (267 МГц) и 40x меньшем количестве вентилях (2.5 млн.)

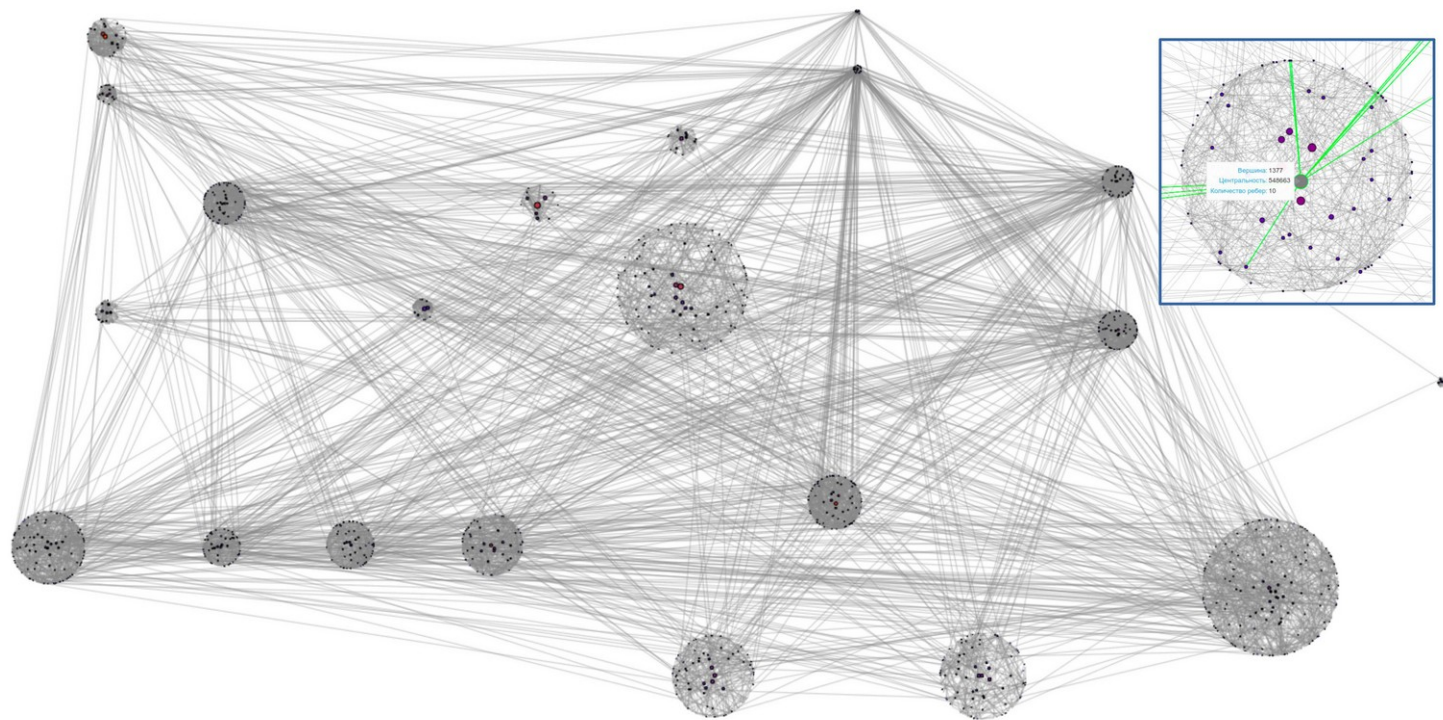
Микропроцессор Леонард Эйлер

- Ядра GPC объединяются в группы ядер (до 6 ядер в группе)
- В каждой группе предусмотрена глобальная память 128КБ для обмена данными между хост-подсистемой и CPE.
- В каждом ядре CPE предусмотрены аппаратные очереди сообщений Host2GPC и GPC2Host на 512 записей по 32 бит каждая.
- Все GPC в одной группе подключены к одной шине памяти DDR4 (16ГБ).
- Хост-подсистема может независимо управлять каждым GPC в отдельности.
- Основным программным компонентом программного ядра является обработчик (подобно шейдеру), который написан на языке C и загружается по запросу хост-системы.

Примеры, исходные коды библиотек: <https://alexbmstu.github.io/2022>



Пример работы одного гетерогенного ядра Тераграф



Определение сообществ и центральности
~4K вершин, 16 млн кратчайших путей

Студенческий практикум по обработке и визуализации графов

Практикум №1. Разработка и отладка программ в вычислительном комплексе Тераграф с помощью библиотеки leonhard x64 xrt

- Изучение принципов работы системы Тераграф
- Индивидуальные задания

Практикум №2. Обработка и визуализация графов в вычислительном комплексе Тераграф

- Изучение алгоритмов обработки и визуализации графов
- Индивидуальные задания

Командный практикум. Обработка и визуализация графов в вычислительном комплексе Тераграф

- Командная разработка (команда от 5 до 10 человек)
- Результат работы команды: визуализация графа по выбранной предметной области (pdf, 300 dpi)

Страница практикума

<https://alexbmstu.github.io/2022/>

Студенческий практикум МГТУ им. Н.Э.Баумана по обработке и визуализации графов

Московский государственный технический университет им. Н.Э.Баумана, 31
октября – 4 декабря 2022

Руководители практикума



[Алексей Попов](#)
МГТУ им. Н.Э.Баумана



[Станислав Ибрагимов](#)
МГТУ им. Н.Э.Баумана



[Егор Дубровин](#)
МГТУ им. Н.Э.Баумана



[Ли Цзяцзянь](#)
МГТУ им. Н.Э.Баумана



[Михаил Гейне](#)
МГТУ им. Н.Э.Баумана

Студенческий
практикум МГТУ им.
Н.Э.Баумана по
обработке и
визуализации
графов

Аннотация

1. Графы знаний

1.1. Актуальность создания
эффективных
программных и
аппаратных средств
обработки графов

1.2. Применения графов в
задачах аналитики
данных и искусственном
интеллекте

2. Структура микропроцессора Леонард Эйлер и вычислительного комплекса Тераграф

2.1. Набор команд
дискретной математики

2.2. Структура
вычислительного
комплекса Тераграф

2.2.1. Хост-подсистема

2.2.2. Подсистема
хранения графов

2.2.3. Подсистема
коммутиции узлов

