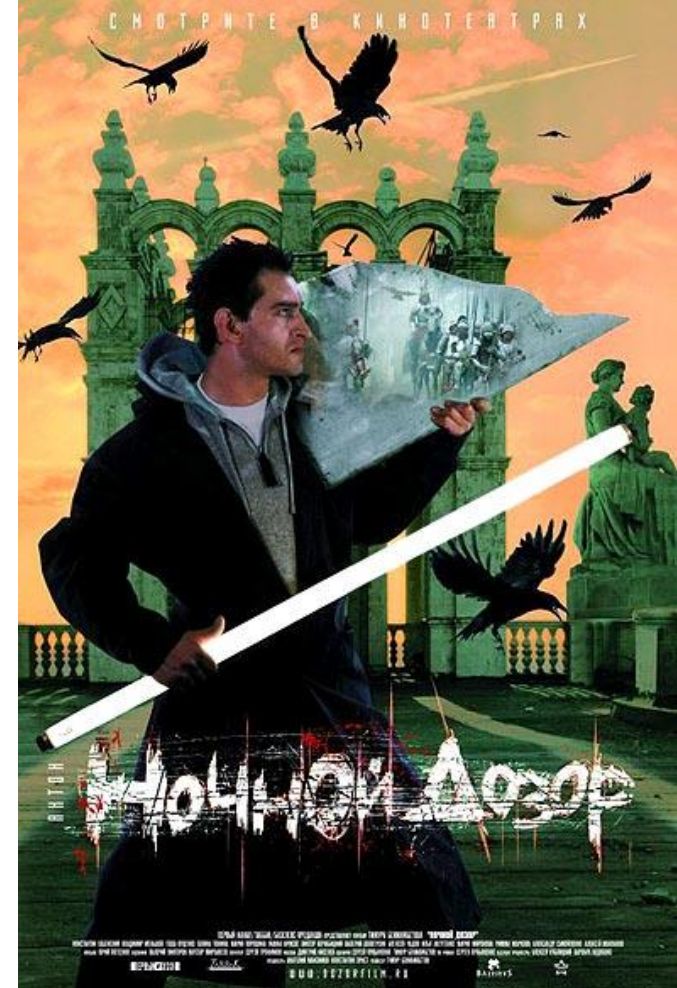


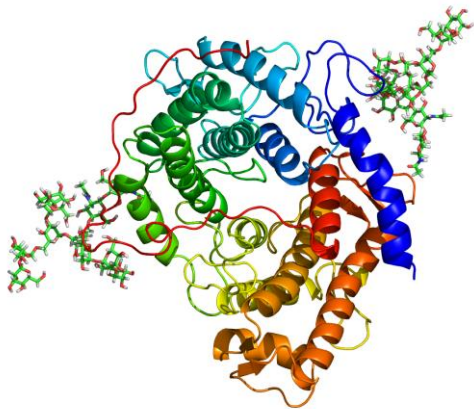
Игорь Одинцов

Группа компаний РСК

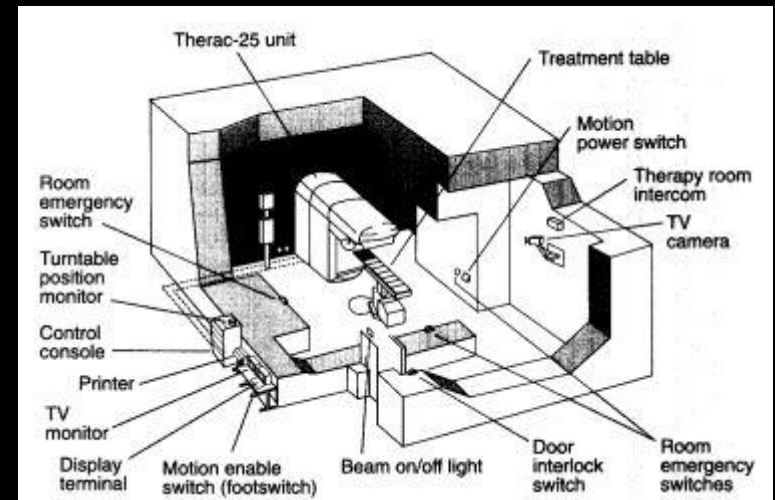
Параллельное программирование
в метафорах:
борьба светлых и тёмных сил



- Производительность
- Результативность
- Нужность



- Ограничения
- Проблемы
- Сложности



Вспоминаем. Что такое метафора?

- **Метафора** – косвенное сообщение в виде истории или образного выражения, использующего сравнение
- Метафоры помогают найти наиболее подходящую аналогию, то есть соединить новый опыт с уже имеющимся
- Используйте метафору, чтобы объяснить что-то «бабушке» (или школьнику!)

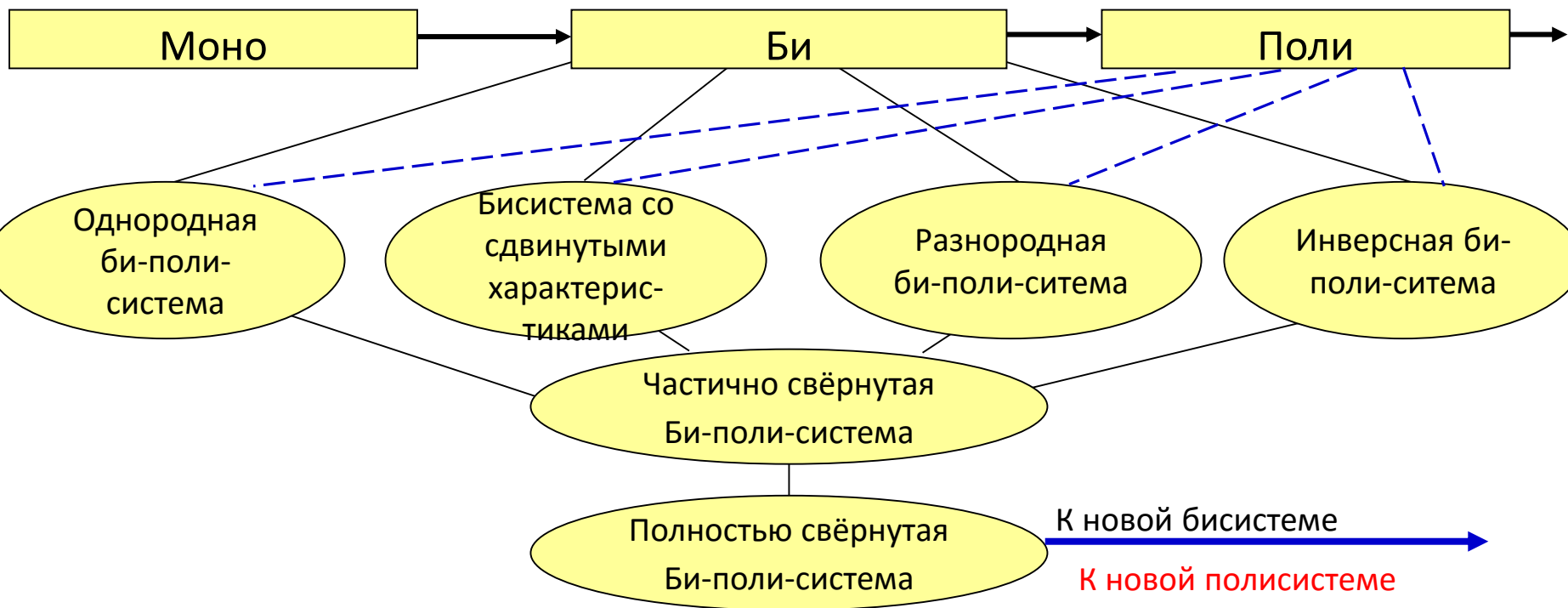
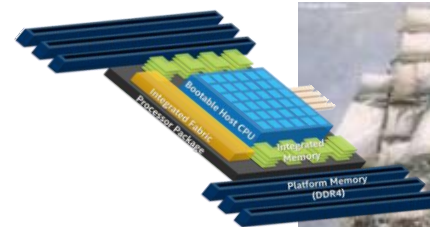
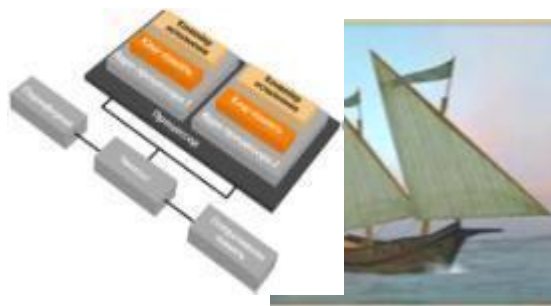
Упражнение №1:

Что такое суперкомпьютер?

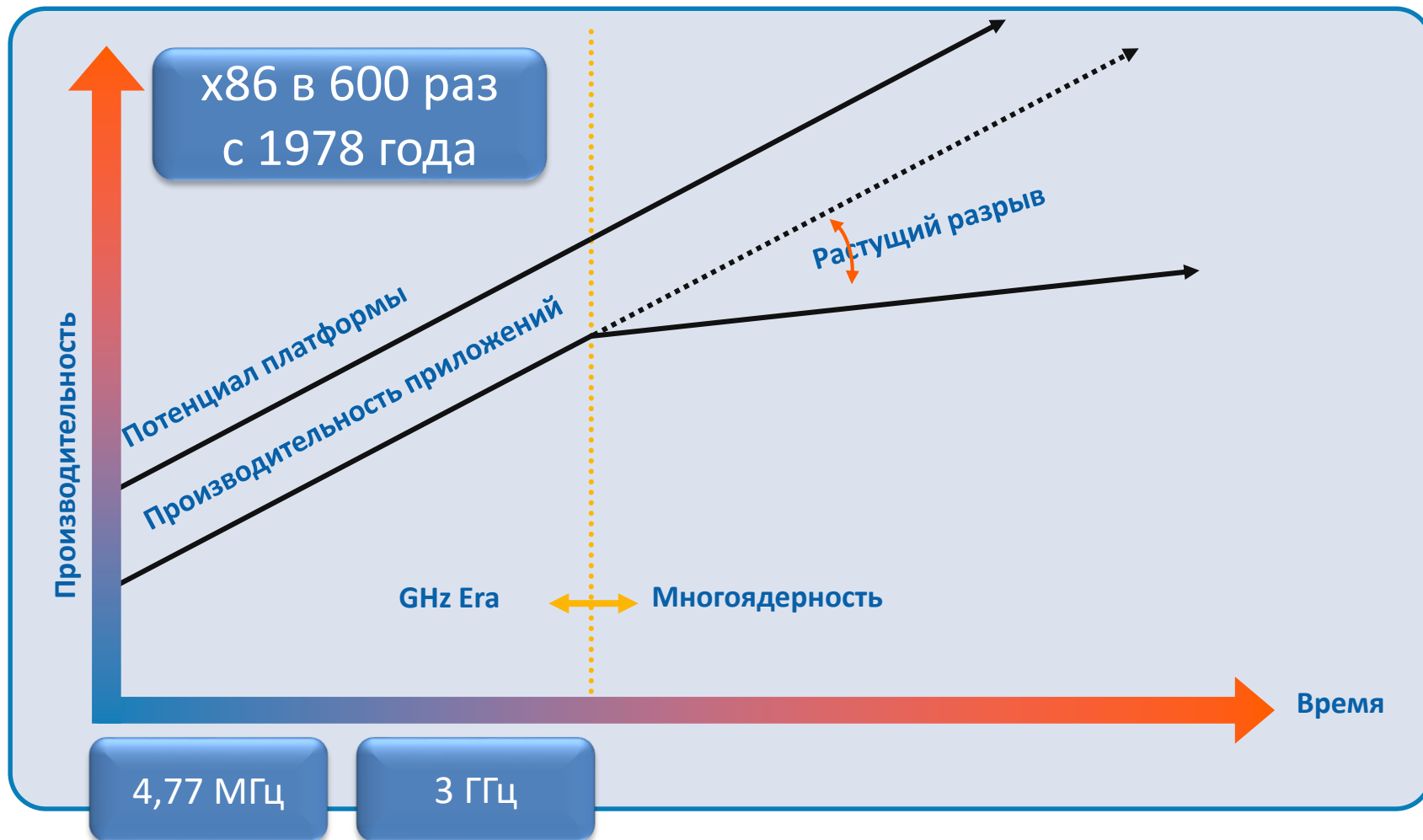
Однажды мне был задан вопрос:

- «Зачем Intel придумал многоядерные процессоры и параллелизм?»
(Подразумеваемая, что без них было просто и хорошо)

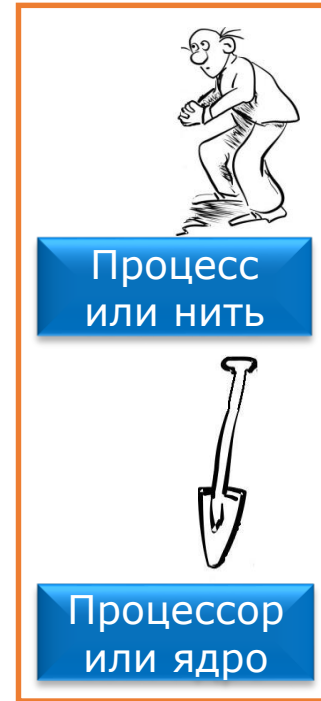
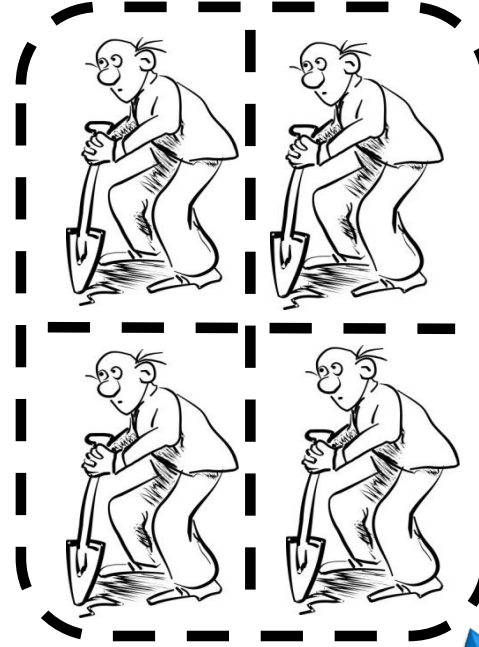
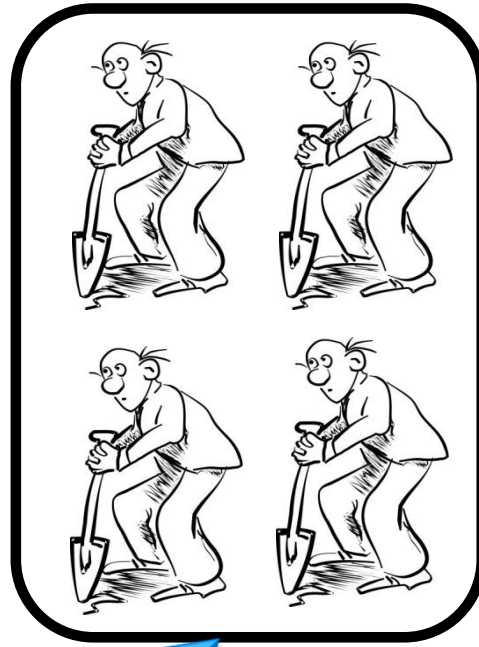
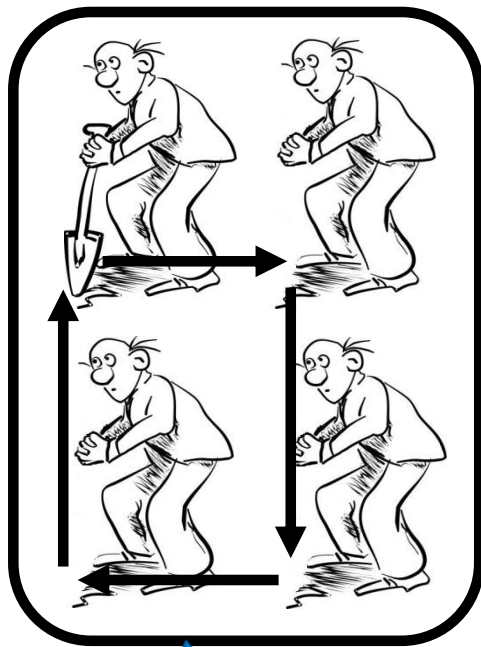
Законы развития технических систем: линия моно-би-поли



Многоядерные процессоры нуждаются в параллельных приложениях



Параллелизм: типы



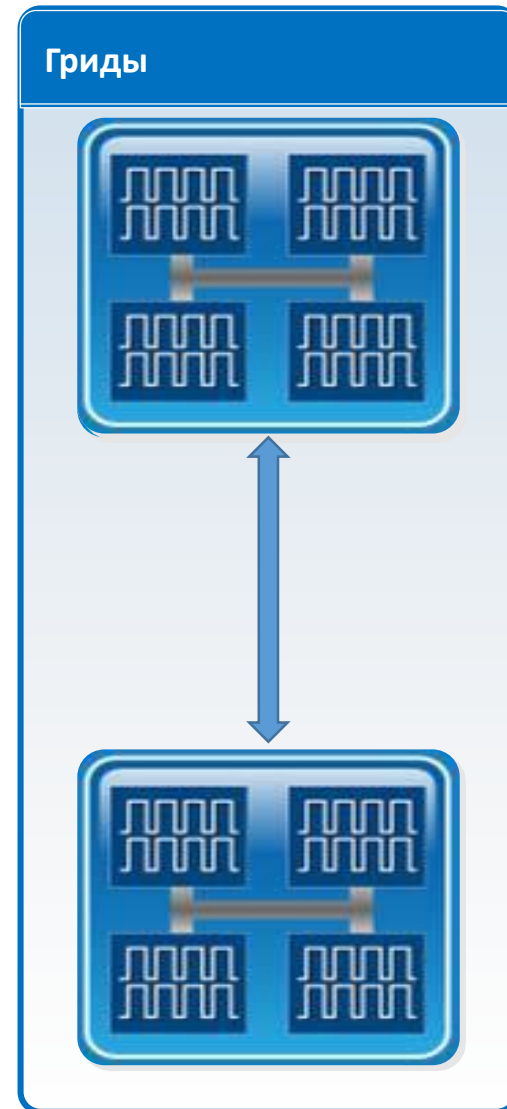
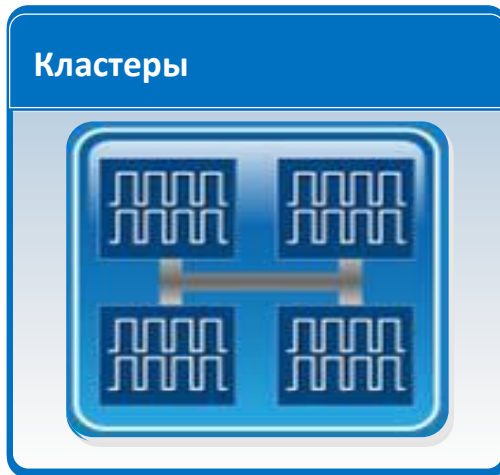
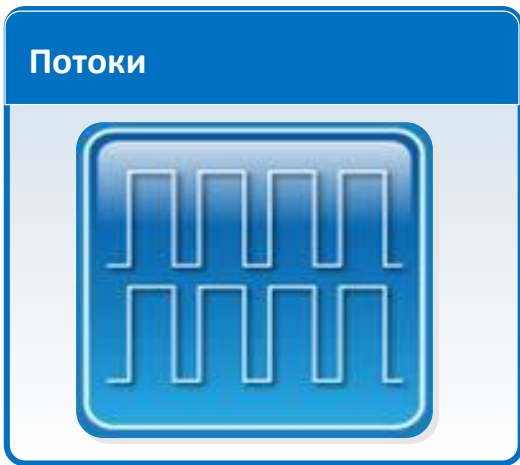
Истинный параллелизм

Псевдопараллелизм

Распределенный параллелизм

- Процессы (нити) называются *параллельными*, если они выполняются одновременно. Они могут быть либо независимыми, либо взаимодействующими и нуждающимися в синхронизации

Параллелизм: взаимодействие



Параллелизм: аппаратный уровень

Векторизация

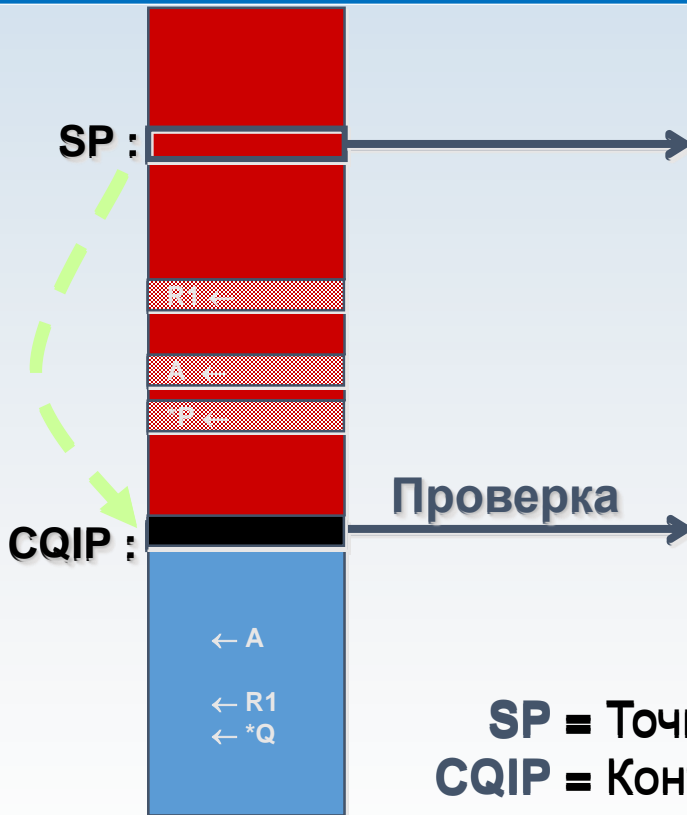
128b

256b

512b

VLIW

Динамический поток команд



Участок предвычислений

- Отклоняются маловероятные ветвления
- Вычисляются входные спекулятивные величины

Упреждающий
поток

SP = Точка ответвления (когда)
CQIP = Контрольная квази-независимая точка (где)

Параллелизм: независимый и с коммуникацией



Photo credit JJ Harrison [\(CC\) BY-SA 3.0](#)

Параллелизм
с коммуникацией



Photo credit André Zehetbauer [\(CC\) BY-SA 2.0](#)

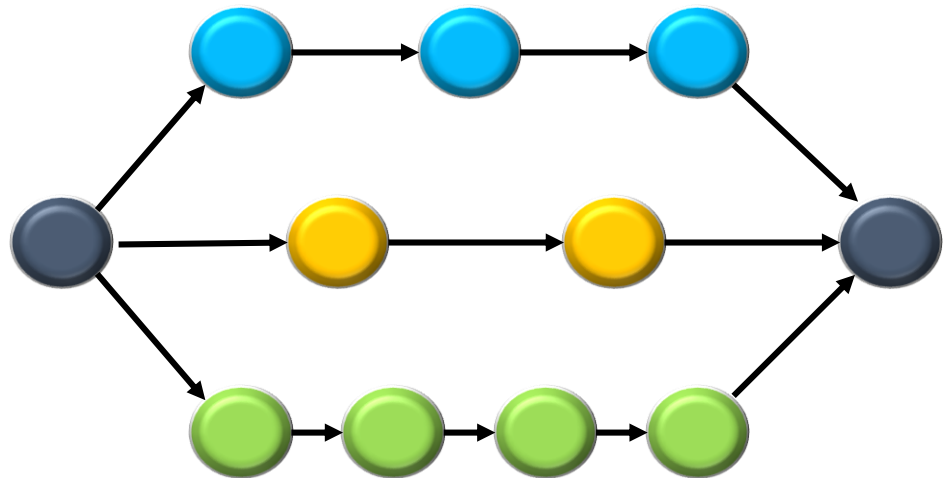
Параллелизм
независимый

Независимый параллелизм

В геометрии
параллельные прямые
не пересекаются:



В программировании
независимые
параллельные задачи
не взаимодействуют



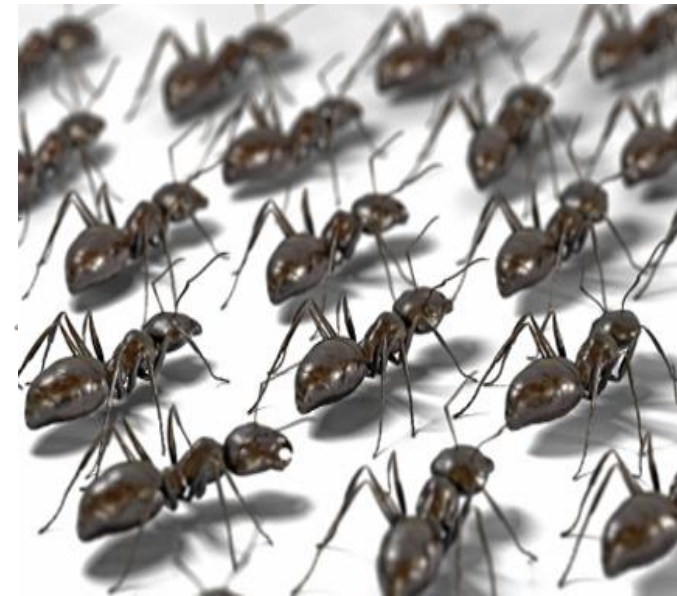
Параллелизм: разновидности задач



Двух
СИЛЬНЫХ
БЫКОВ



...или 1'024 курицы



...или 10'240'560 муравьев?

Ядра:

НЕМНОГО МОЩНЫХ VS. МНОГО ПРОСТЫХ

Десятки ядер

Высокая частота

Супер-
современная
микро-
архитектура



Сотни ядер

Низкая частота

Упрощенная
микро-
архитектура



Упражнение №2: Путь проблемно-ориентированного программирования: от задач к железу

Обл1) Сейсморазведка

Обл2) Гидродинамика

Обл3) Биофизика



Арх1) Intel Xeon

Арх2) Intel Xeon Phi

Арх3) Эльбрус

Проблемно-ориентированное программирование: от задач к железу

Обл1) Сейсморазведка

Обл2) Гидродинамика

Обл3) Биофизика

1.1. Предметные области и задачи

1.2. Вычислительные модели

4. Решение:

- производительность
- энергоэффективность
- стоимость

2.1. Методики управления вычислениями

2.2. Методики управления данными

2.3. Языки:

- Моделирования, Программирования, Библиотек, Ранее написанной функциональности, Оптимизации, Распараллеливания, ...

2.4. Инструменты:

- Среды разработчика,
- Среды исполнения,
- Оптимизации,
- Распараллеливания,
- Системное ПО
- ...

3. Вычислительные архитектуры (платформы и коммуникации)

- ядра-процессоры, память-СХД, коммуникации-компьютерные шины, узлы-кластера, ...

Арх1) Intel Xeon

Арх2) Intel Xeon Phi

Арх3) Эльбрус

Проблемно-ориентированное программирование: от задач к железу (о важности библиотек!)

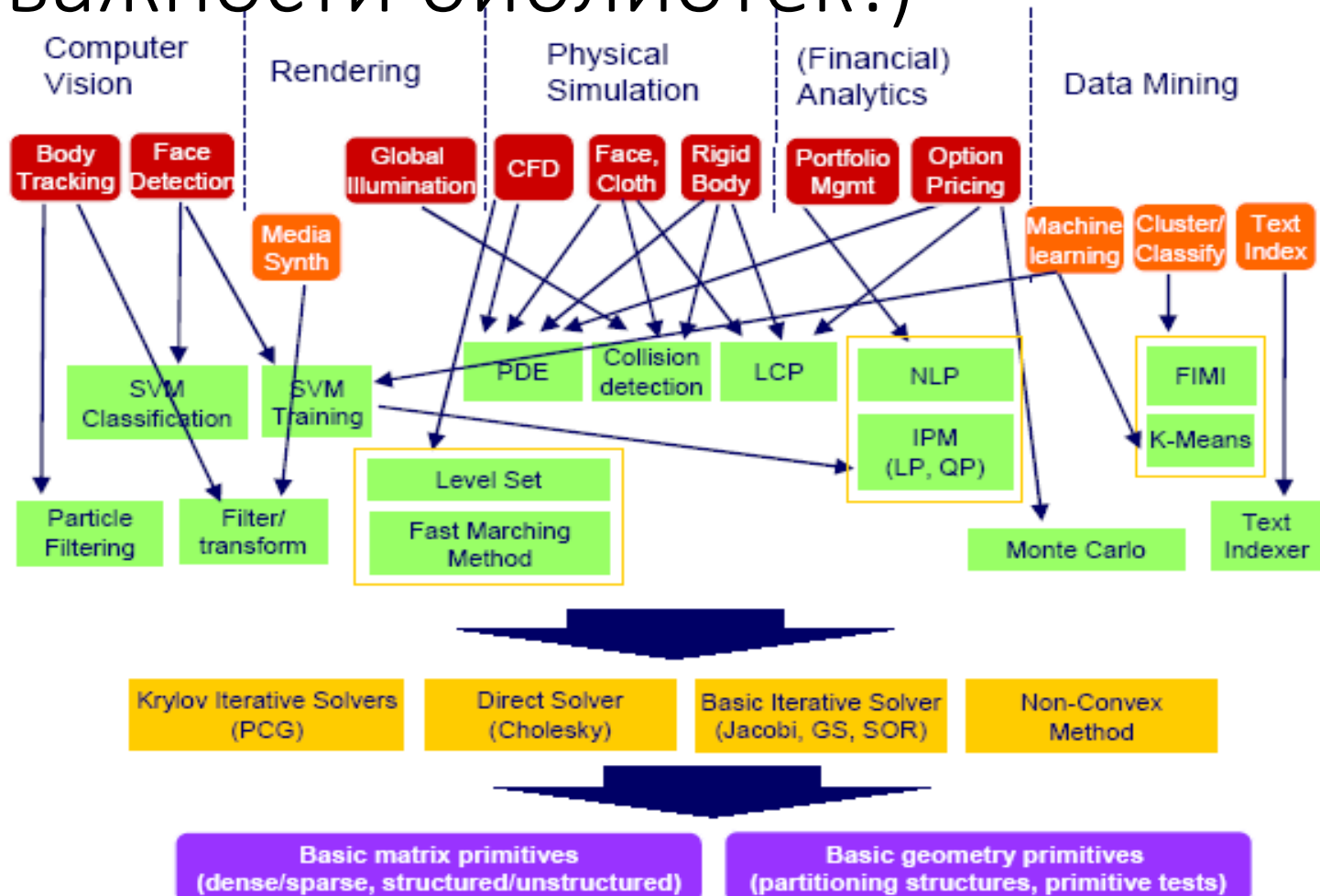
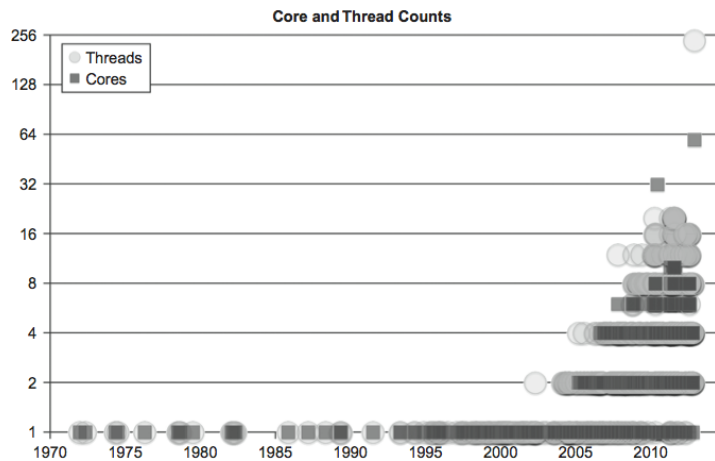
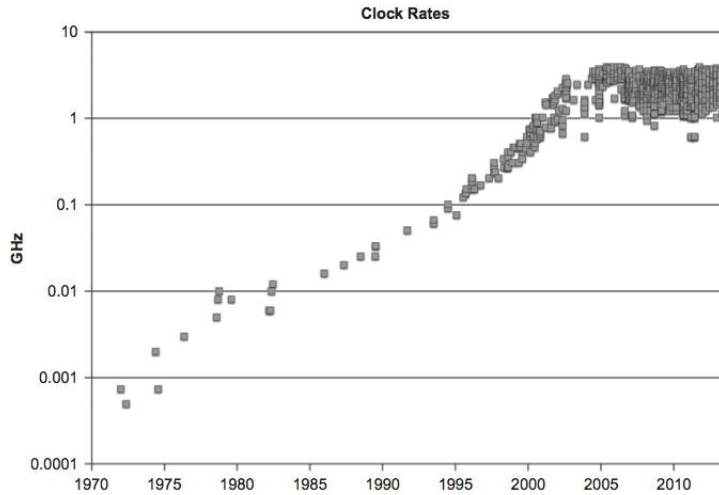


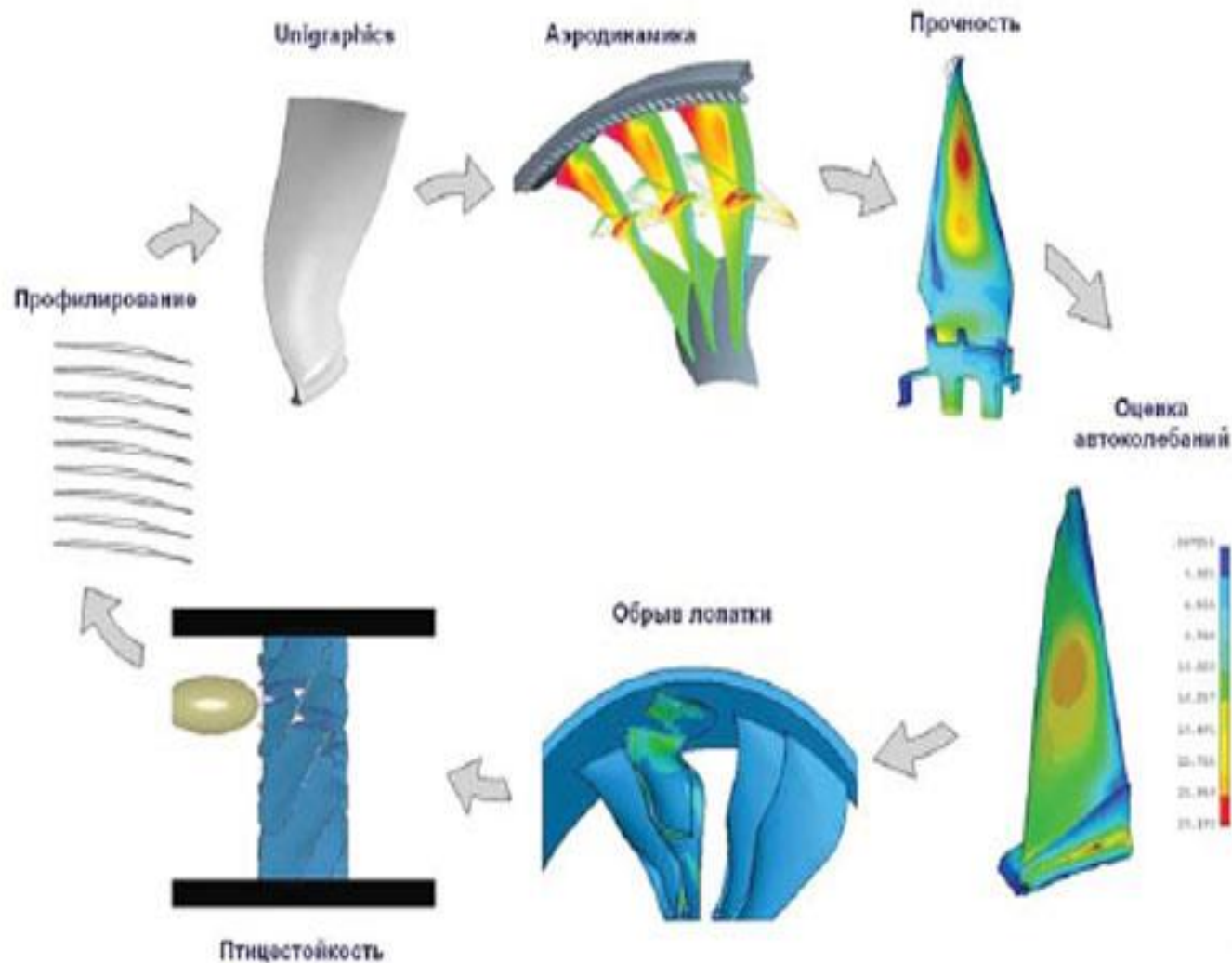
Figure 5. Intel's RMS and how it maps down to functions that are more primitive. Of the five categories at the top of the figure, Computer Vision is classified as Recognition, Data Mining is Mining, and Rendering, Physical Simulation, and Financial Analytics are Synthesis. [Chen 2006]

Параллелизм – норма жизни

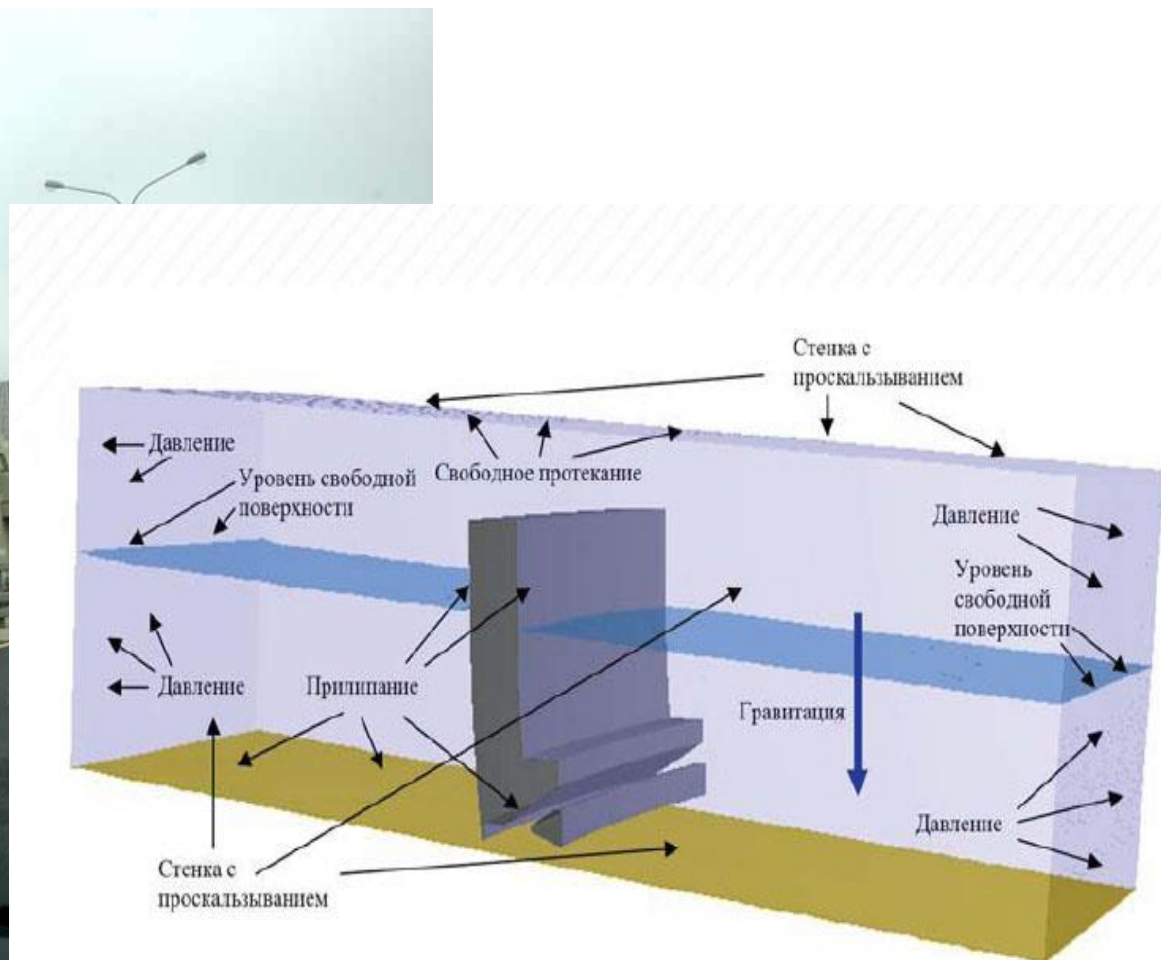


- Мультитядерные и многоядерные процессоры
- Степень аппаратного параллелизма продолжает расти
- Требуется явный параллелизм на уровне софта

Моделирование в газодинамике

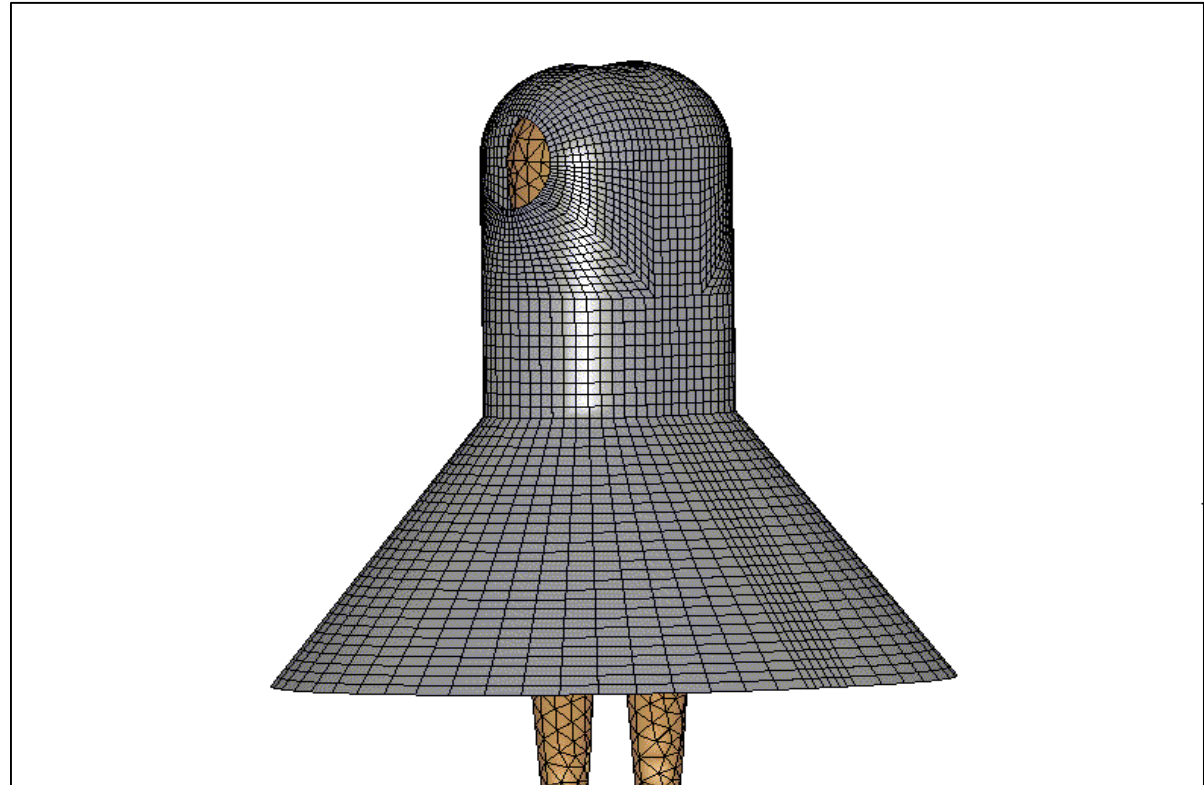
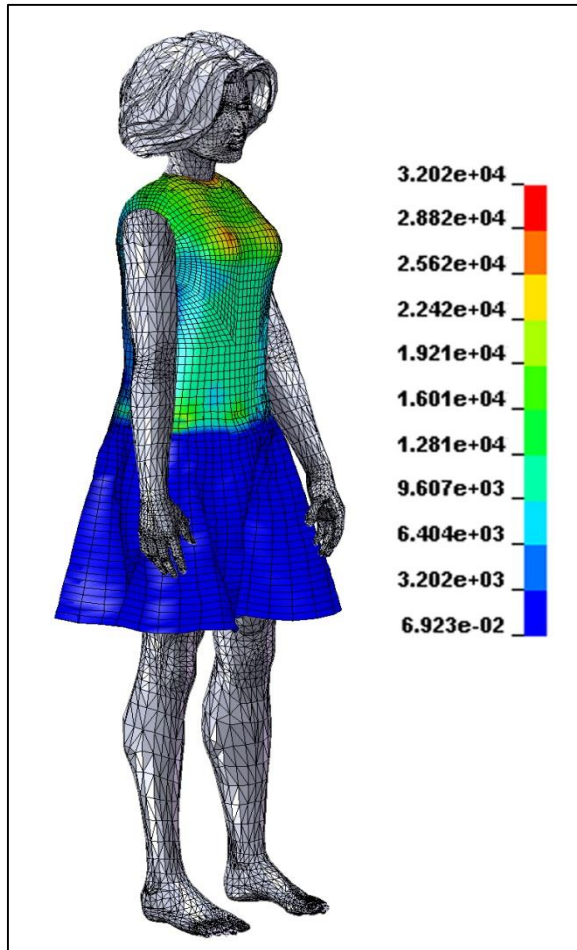


Моделирование в гидродинамике

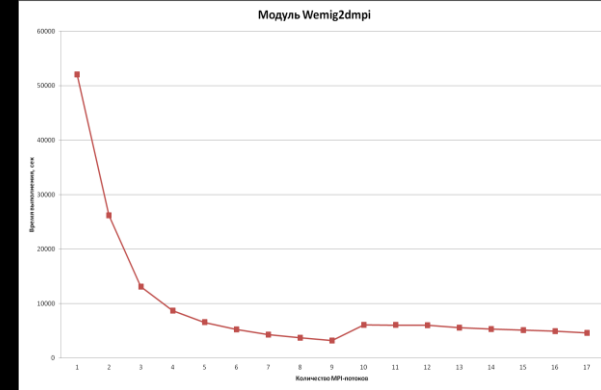


Трёхмерная модель батопорта

Механизм «одевания» виртуального платья на манекен

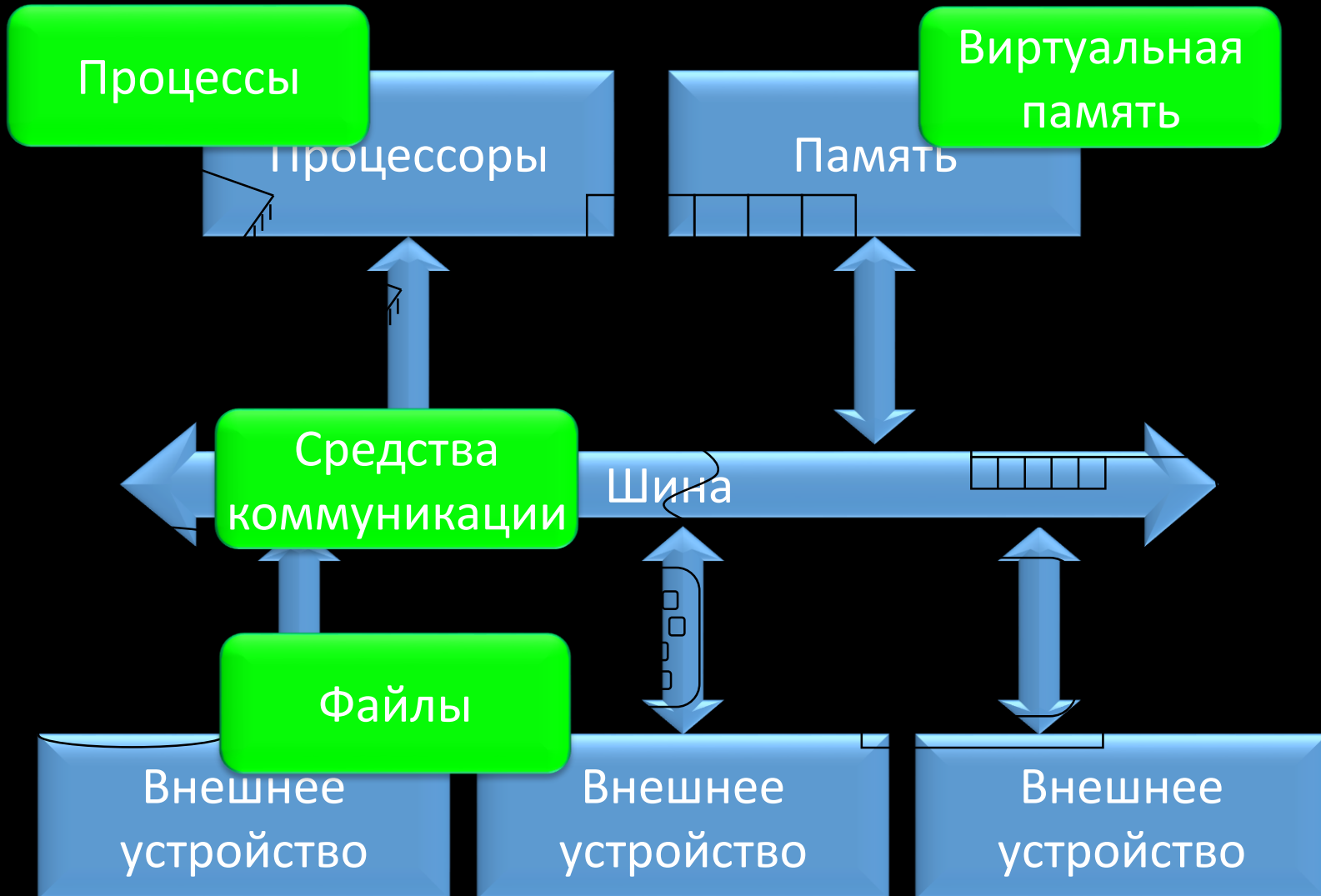


Закон Амдала



- «В случае, когда задача разделяется на несколько частей, суммарное время её выполнения на параллельной системе не может быть меньше времени выполнения самого длинного фрагмента»
- Закон Амдала показывает, что прирост эффективности вычислений зависит от алгоритма задачи и ограничен сверху для любой задачи
- Не для всякой задачи имеет смысл наращивание числа процессоров в вычислительной системе

Вечная борьба за ресурсы: аппаратные и программные («и пряников, кстати, всегда не хватает на всех» (с))



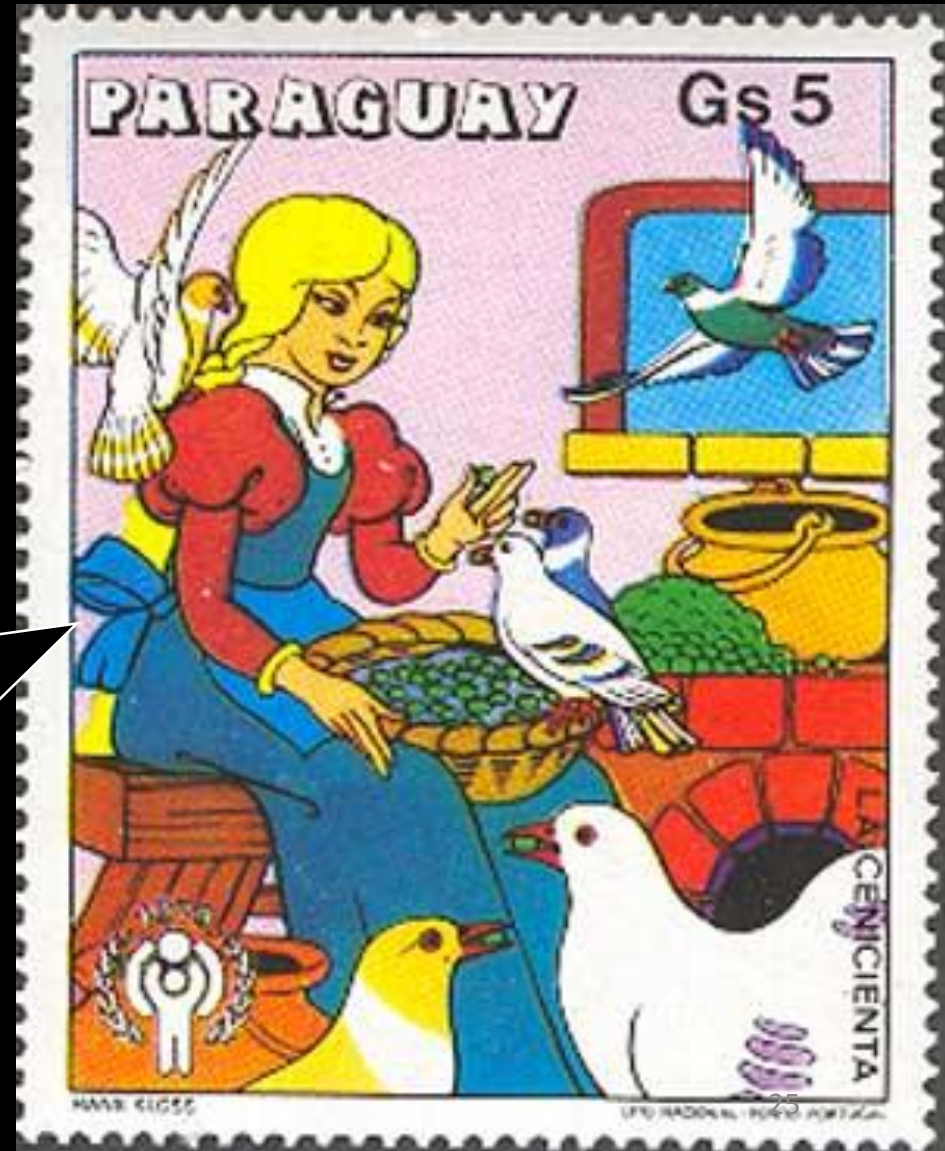
Упражнение №3: Почему программы зависают?

- Борьба за ресурсы!
Пример: круговое ожидание

Почему я перестал верить в сказки...



А где находится
подсистема
коммуникации и
синхронизации
птичек и др.?



Упражнение №4:

И всё-таки - легко ли писать параллельные программы?

- Пример: счетчик

Кстати, давно придуманы красивые решения, например – семафоры

- Семафор — это защищенная переменная, значение которой можно запрашивать и менять только при помощи специальных операций P и V и при инициализации
- Концепция семафоров была предложена Дэйкстрой в начале 60-х годов XX века

P(S):

if ($S > 0$) then

$S := S - 1$

else

 ожидать_в_очереди(S)

V(S):

if (есть_процессы_в_очереди(S)) then

 одному_продолжить (S)

else

$S := S + 1$



Параллелизм – это сложно

«Кто виноват?»

Слишком низкий уровень абстракции

Отсутствие необходимых знаний и опыта

Некоторые концепции **действительно** сложны

«Что делать?»

Нанять эксперта в разработке параллельных программ

Стать таким экспертом

Использовать другие подходы к параллелизму, например – библиотеки параллельных примитивов

Так не должно быть!
(это нелепо и неэффективно)



А вот так - не идеально,
но гораздо лучше! 😊



Упражнение №5 (более сложное)
Определите OpenMP и MPI метафорически
на примере дорожного движения

Спасибо!

