Технология Фрагментированного Программирования

Летняя школа по параллельному программированию Отдел МО ВВС ИВМиМГ СО РАН 2009 г.

ФП – технология параллельного программирования

 Вопрос: зачем нужна ещё одна технология параллельного программирования?

• Ответ: Существующие технологии не справляются со своей задачей

Что мы хотим от технологии ПП?

 Хотим писать программы с минимумом усилий, а в результате получать эффективную программу

• Такого не бывает!

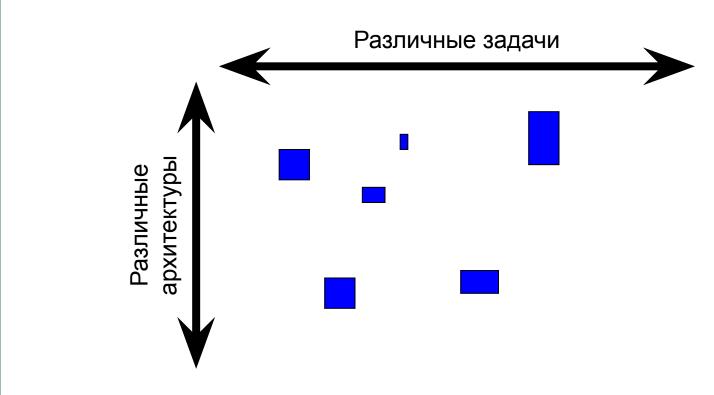
Синтез программ

 Идея: синтезировать программу из готовых программ

 Вопрос: почему до сих пор никто такого не сделал?

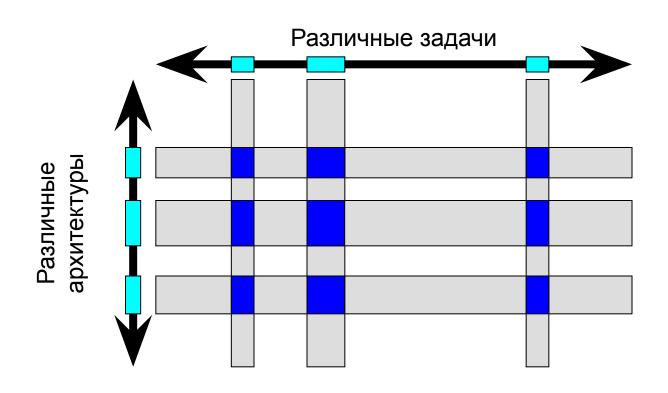
 Ответ: программы плохо совместимы друг с другом!

Плохая совместимость программ



Вопрос: Что делать?

Разделение обязанностей



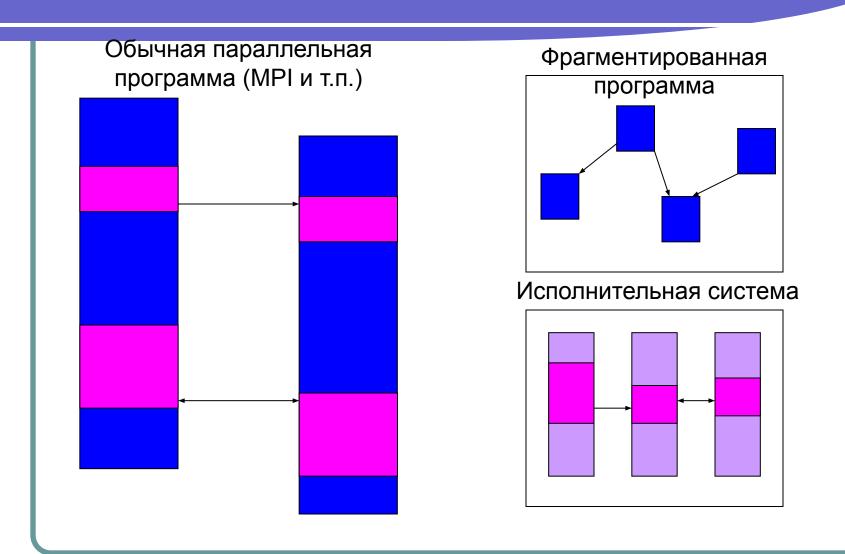
Представление алгоритма

Задачи Исполнительной Системы

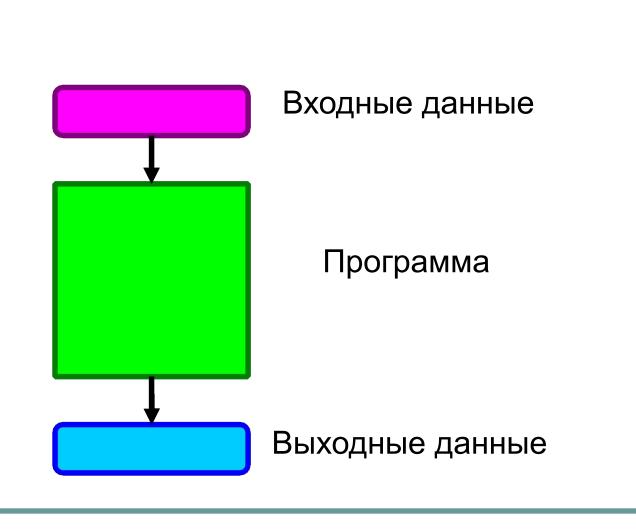
- Явный параллелизм
- Структура данных и вычислений
- Гибкая параметризация
- Возможность сборки
- Использование обычных подпрограмм
- Рекомендации по способу исполнения

- Настройка на аппаратные ресурсы
- Управление вычислениями и коммуникациями
- Динамическая
 балансировка загрузки
 - Run-time оптимизация

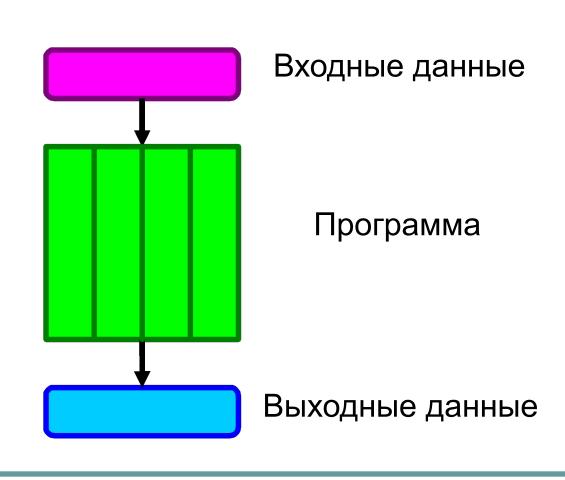
Исполнительная система



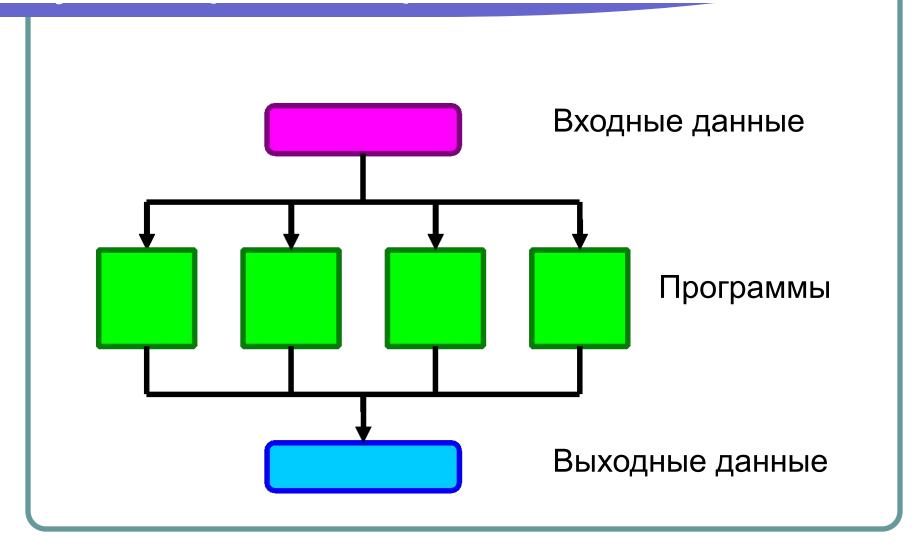
Последовательное программирование



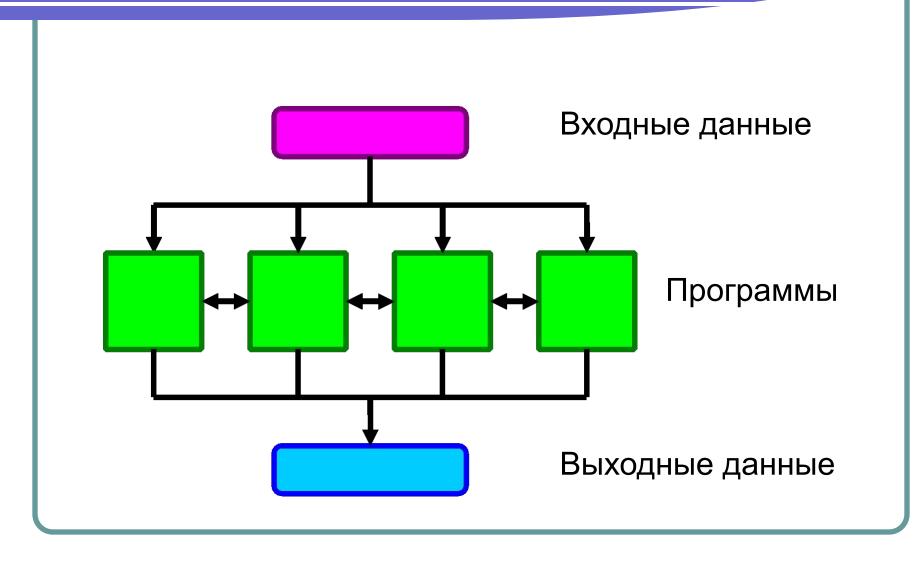
Декомпозиция вычислений



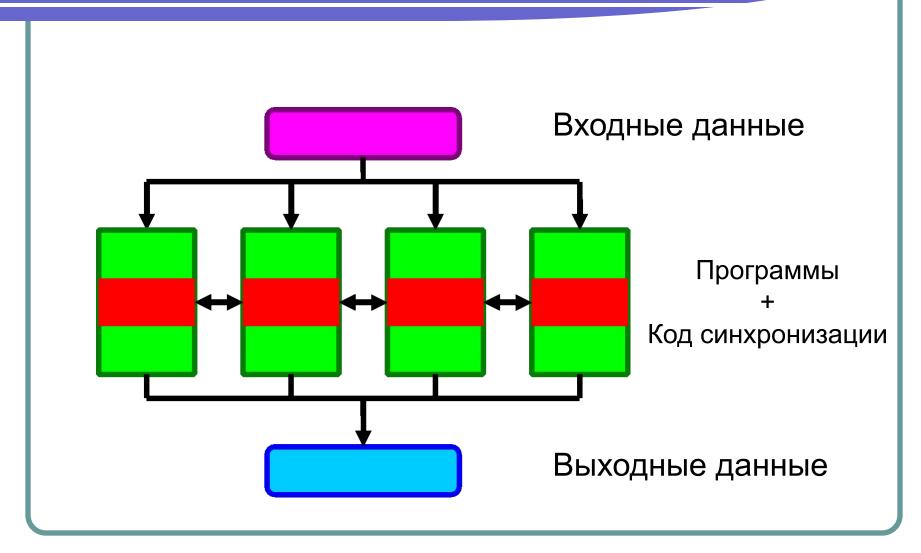
Параллельное исполнение на мультипроцессоре



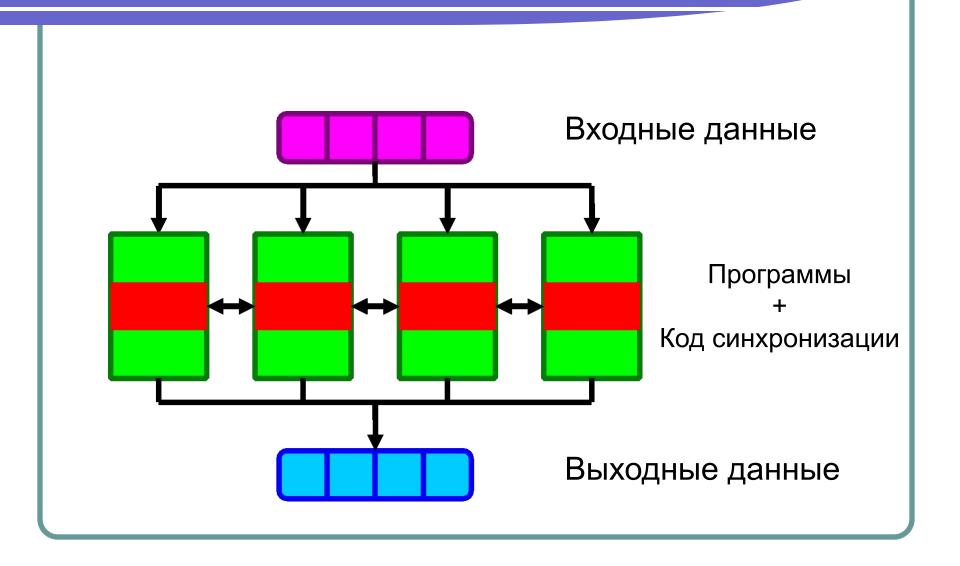
Синхронизация потоков исполнения



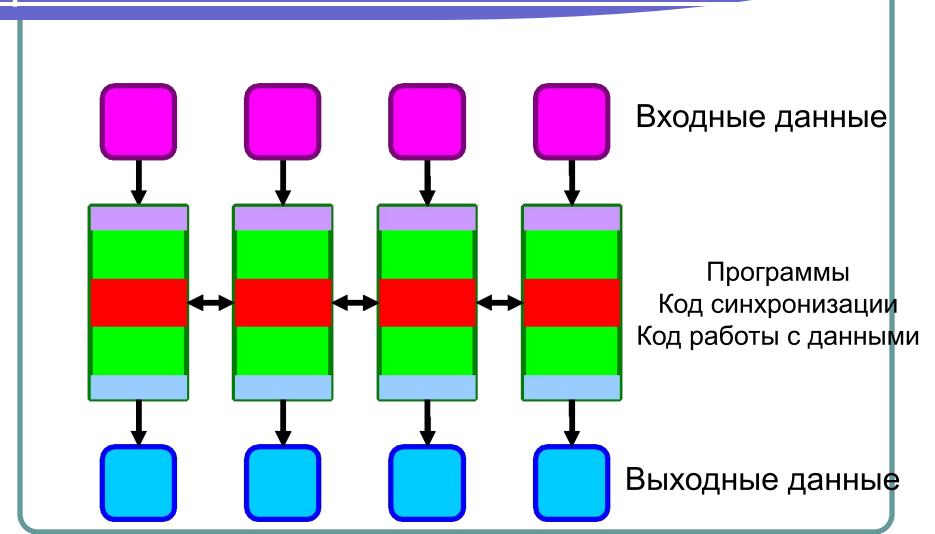
Синхронизация потоков исполнения



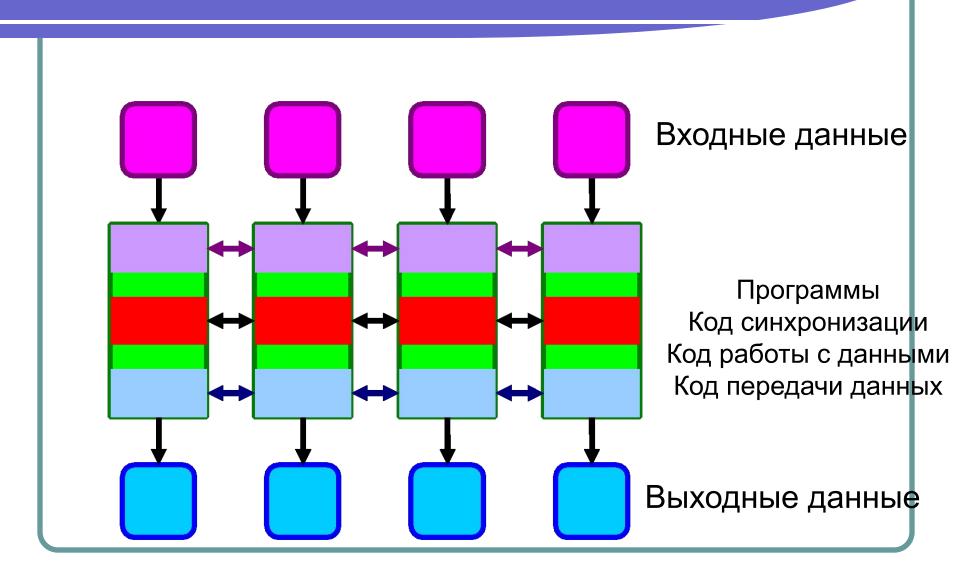
Декомпозиция данных



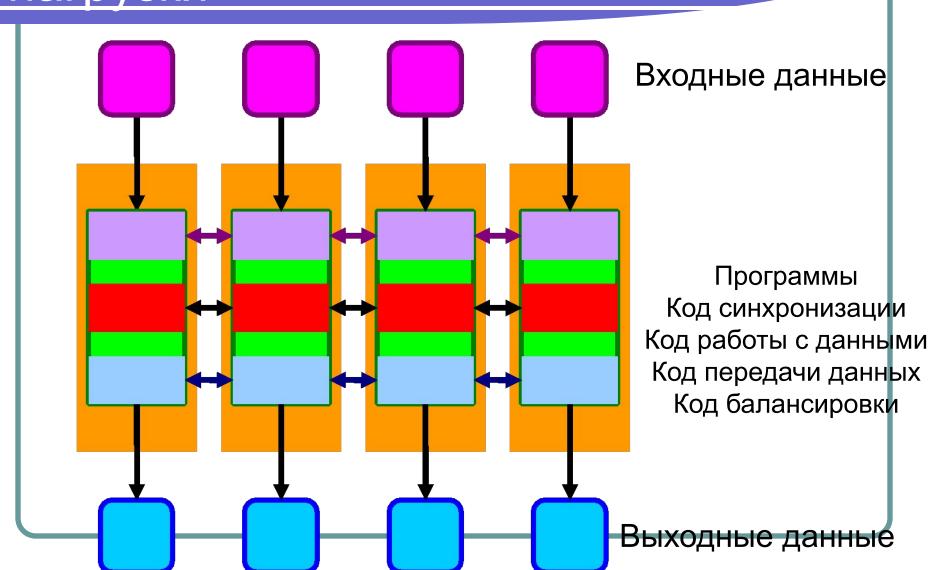
Дополнительные данные и код работы с ними



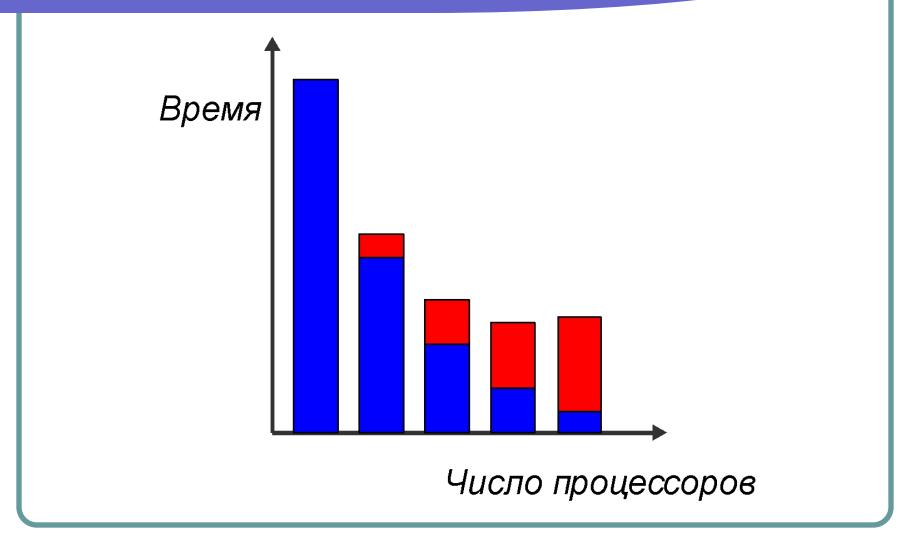
Динамическая передача данных



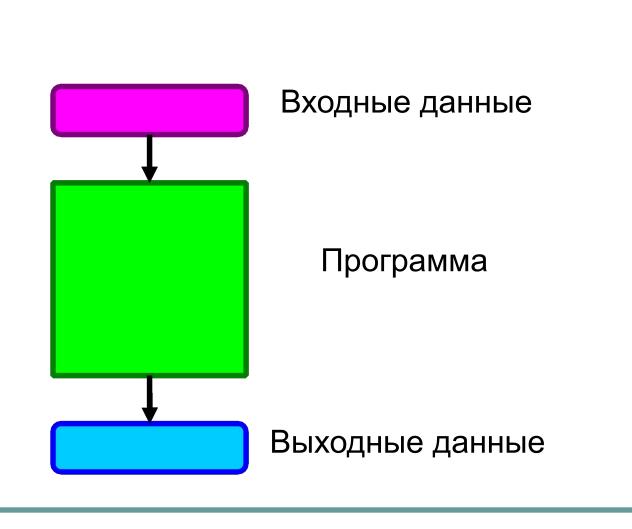
Динамическая балансировка нагрузки



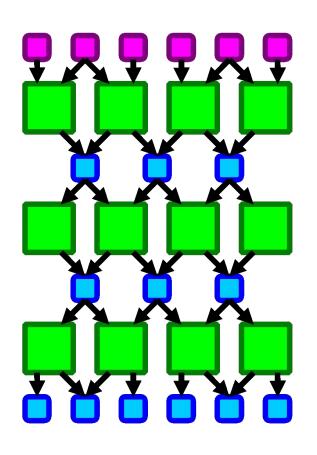
Накладные расходы в параллельных задачах



Последовательное программирование



Фрагментированная программа

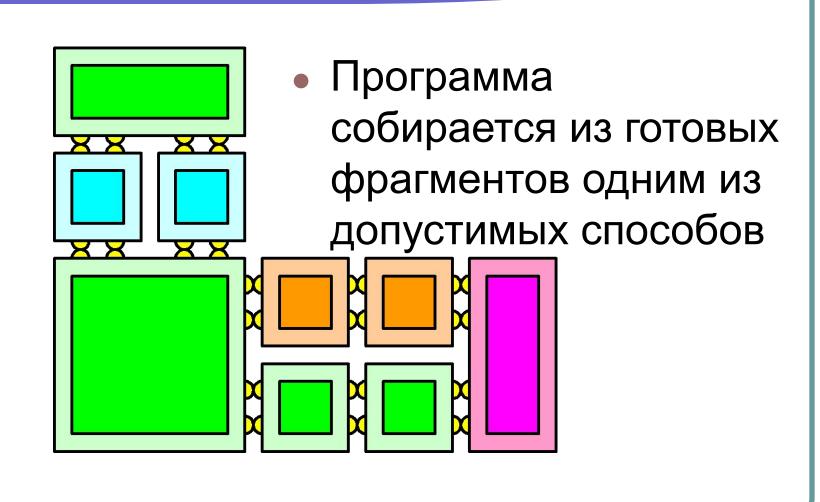


Фрагментированные входные данные

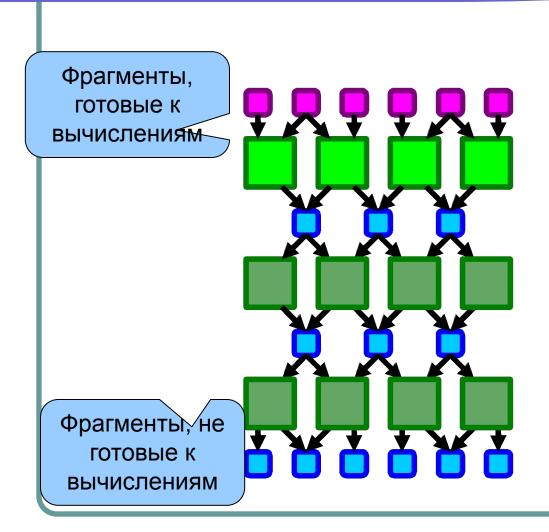
Фрагментированная Программа и промежуточные данные

Фрагментированные выходные данные

Сборка программы из фрагментов



Фрагментированная программа в процессе исполнения



Фрагментированная программа в процессе исполнения

TYA TYA

Фрагменты, готовые к вычислениям

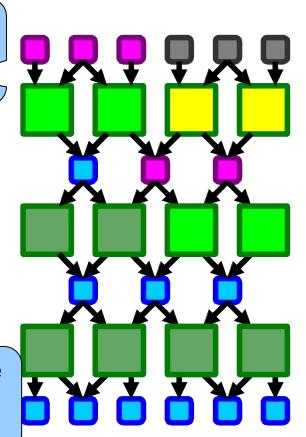
Выполненные фрагменты

Вычисленные значения

Фрагменты, не готовые к вычислениям

Фрагментированная программа в процессе исполнения

Фрагменты, готовые к вычислениям



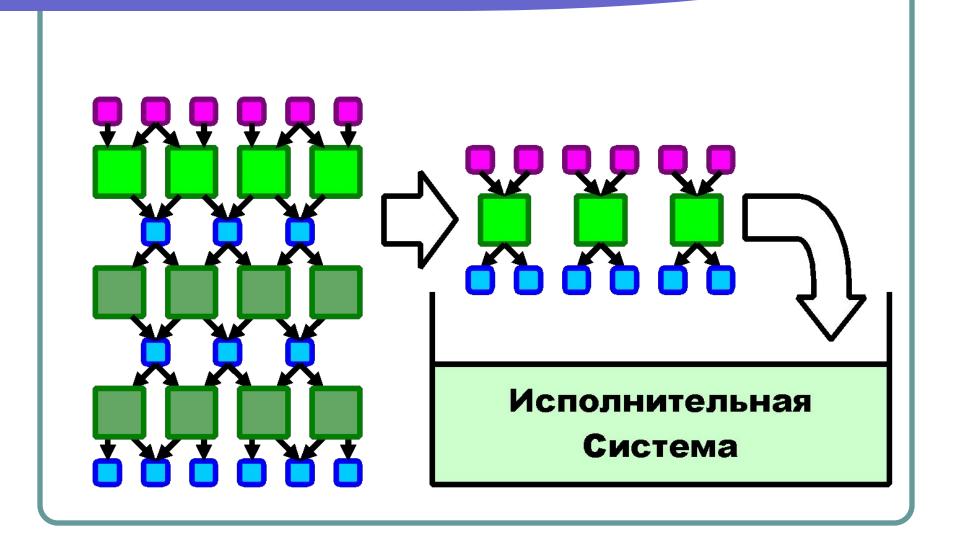
Ненужные более данные

Выполненные фрагменты

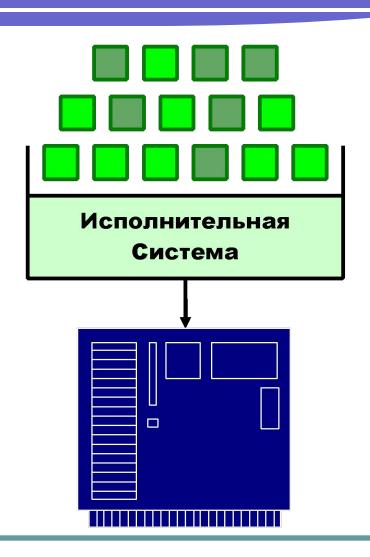
Вычисленные значения

Фрагменты, не готовые к вычислениям

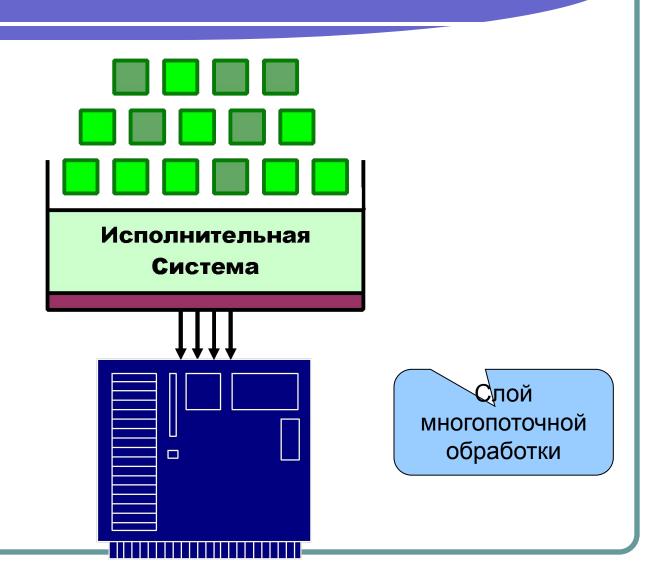
Исполнительная система



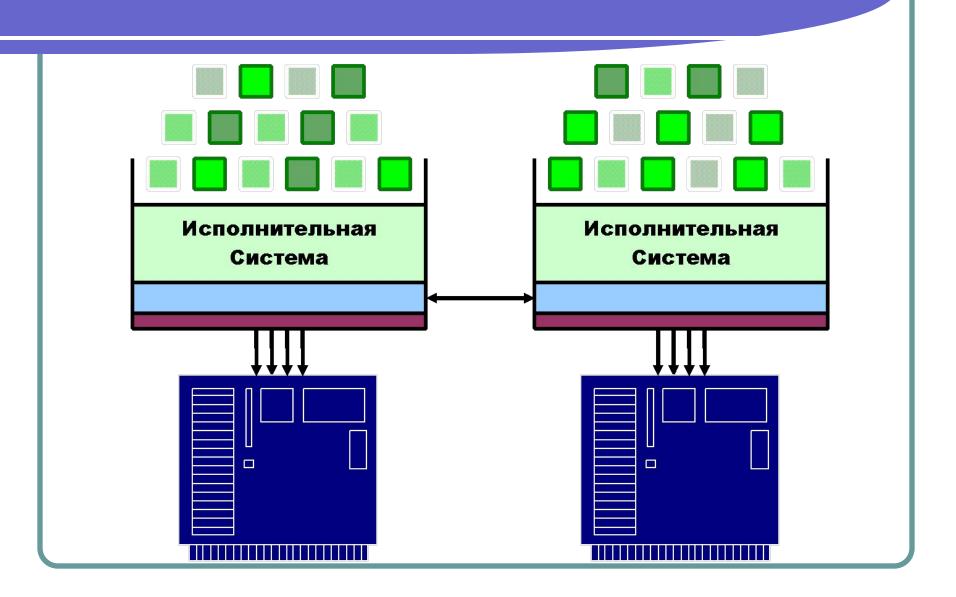
Последовательная ИС



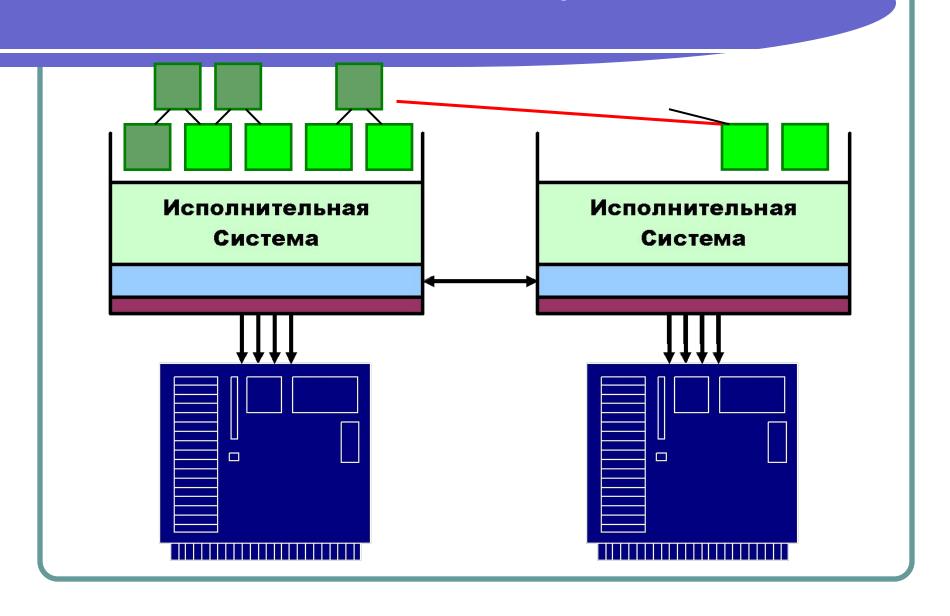
Мультипроцессорная ИС



Мультикомпьютерная ИС

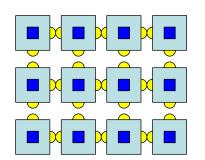


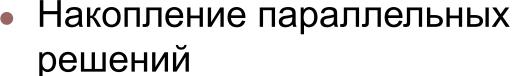
Балансировка нагрузки



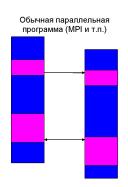
Особенности фрагментированного программирования

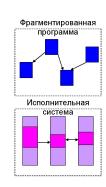
- Явный параллелизм
- Исполнительная система
 - Планировка вычислений
 - Балансировка нагрузки
 - Коммуникации на фоне
 - счета





Высокоуровневое описание алгоритма

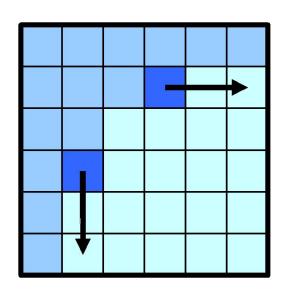


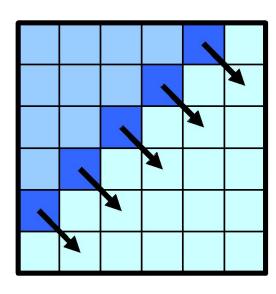


Модель: Фрагментированная программа

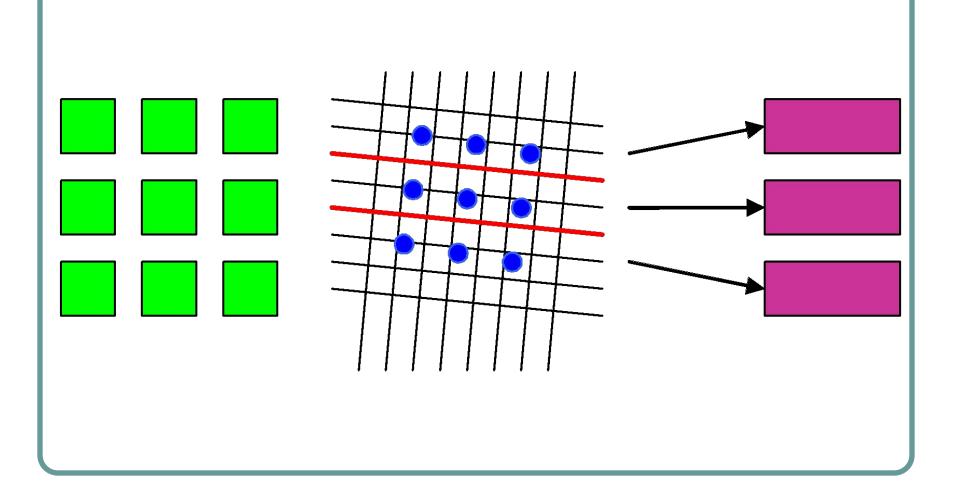
- Фрагменты вычислений и данных
- Формальные и фактические фрагменты
- Структура данных и вычислений
- Управление: прямое, потоковое
- Переменные программы vs алгоритма
- Способ исполнения: какие бывают рекомендации

Разный порядок исполнения

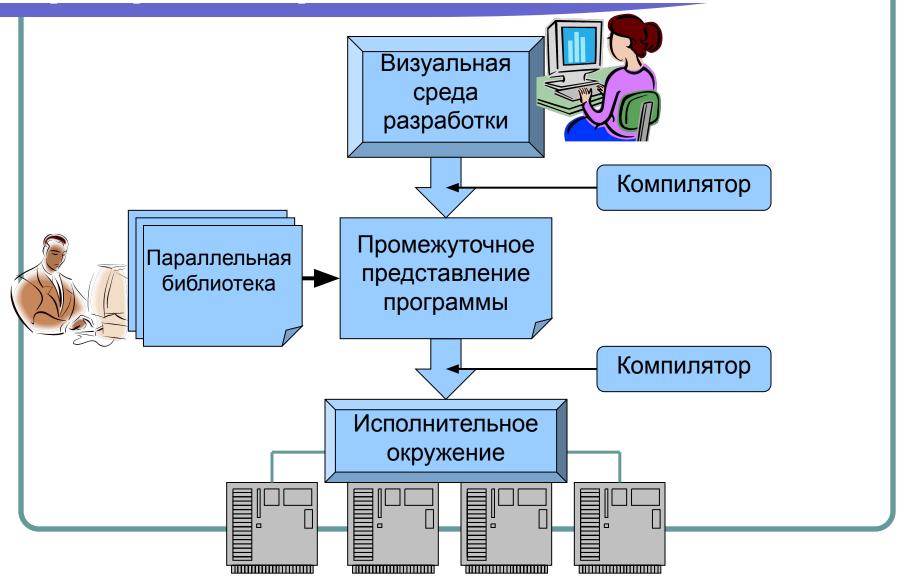




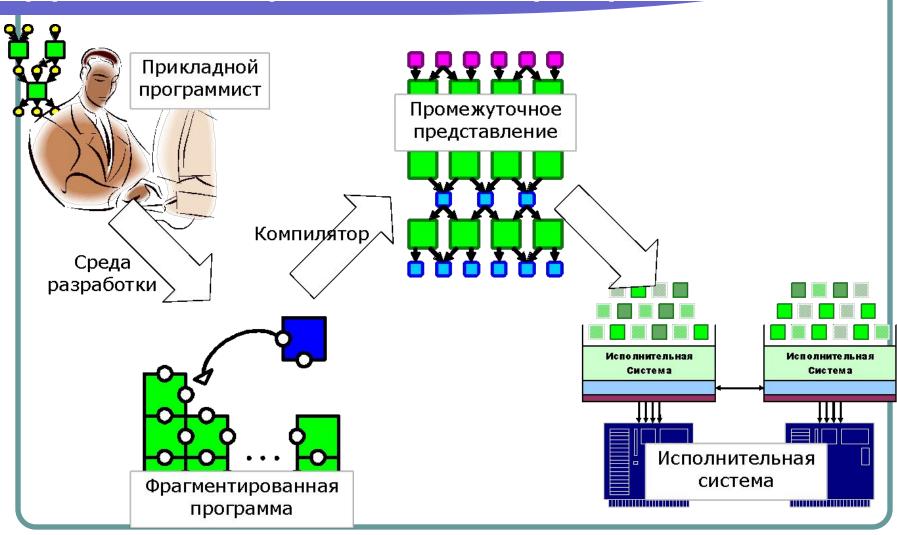
Структурная составляющая



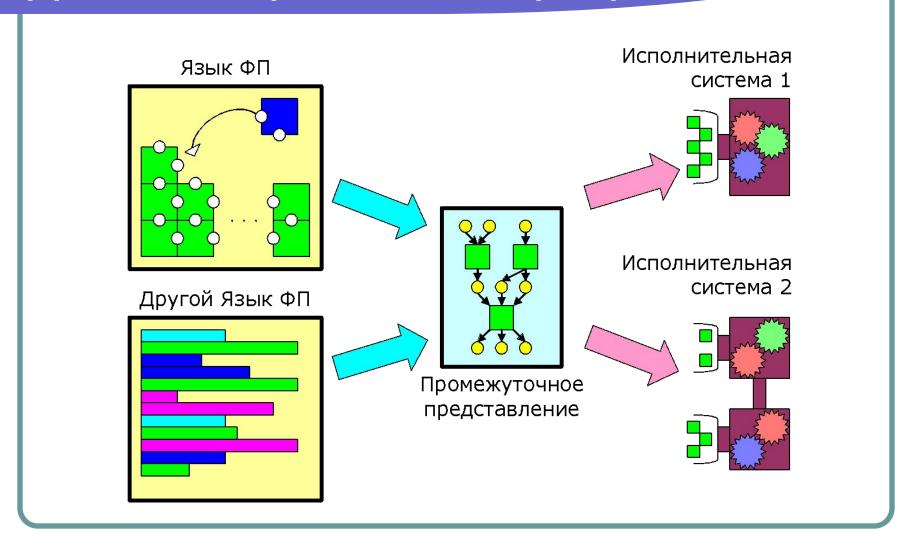
Система Фрагментированного Программирования



Жизненный цикл фрагментированной программы



Промежуточное представление фрагментированной программы



Обзор проекта ФП

- Языки высокого уровня Текстовый
 - Визуальный
- Язык машинного представления
- Исполнительные системы
 - Для мультипроцессоров
 - Для мультикомпьютеров
 - Для спецвычислителей (Cell BE, GPGPU)
- Библиотеки подпрограмм

Технология Фрагментированного Программирования

Конец ⊙