

Летняя международная XXIX молодёжная
Школа-конференция по параллельному программированию

Управление динамическим подключением процессов

Мошкина А.Д., Нестёркина А.А.

— Руководитель: Городничев М.А. —

03.07.2017-14.07.2017

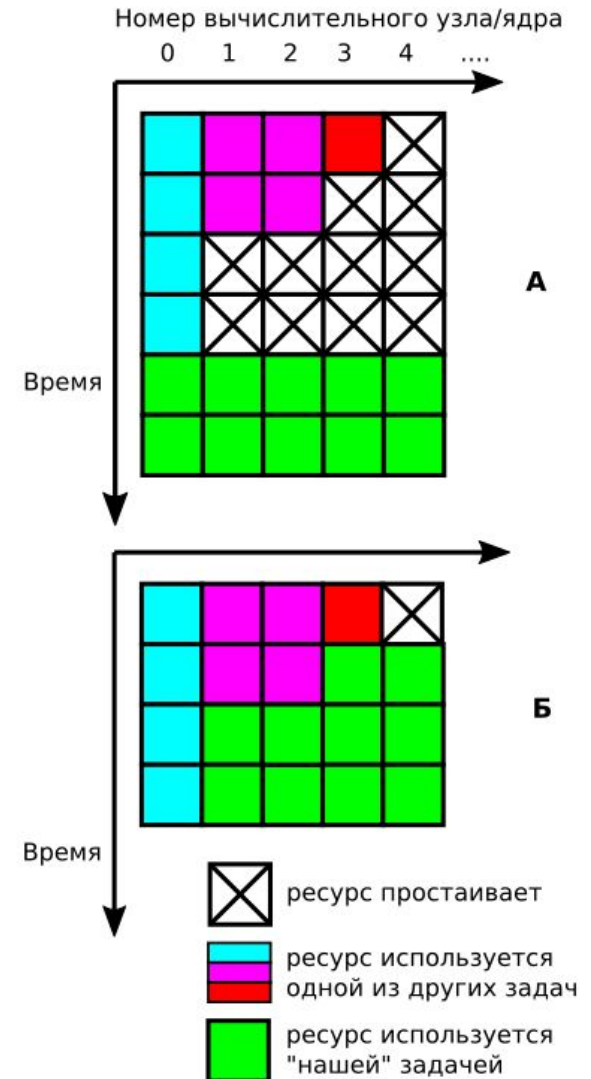
План

1. Цель
2. Идея решения
3. Схемы взаимодействия процессов
4. Программные средства и архитектура
5. Заключение
6. Дальнейшая работа

Цель

Проблема: неэффективное использование ресурсов высокопроизводительных вычислительных систем.

Цель: разработка библиотеки для управления распределением вычислений в системе с динамическим изменением доступности ресурсов.



Идея решения

Два подхода:

1. Статистическое планирование

количество решаемых задач, их объём и количество ресурсов известно заранее и есть возможность некоторым образом распределить данные задачи между ядрами

2. Динамическое распределение

количество решаемых задач заранее неизвестно или неизвестно время выполнения задач, или неизвестно количество ресурсов, которые динамически меняются

Идея решения

Статистическое планирование

Преимущества: определение максимально эффективного решения, временные затраты на планирование однократны.

Недостатки: сужение класса задач, временные затраты на поиск эффективного распределения ресурсов.

Идея решения

Динамическое распределение

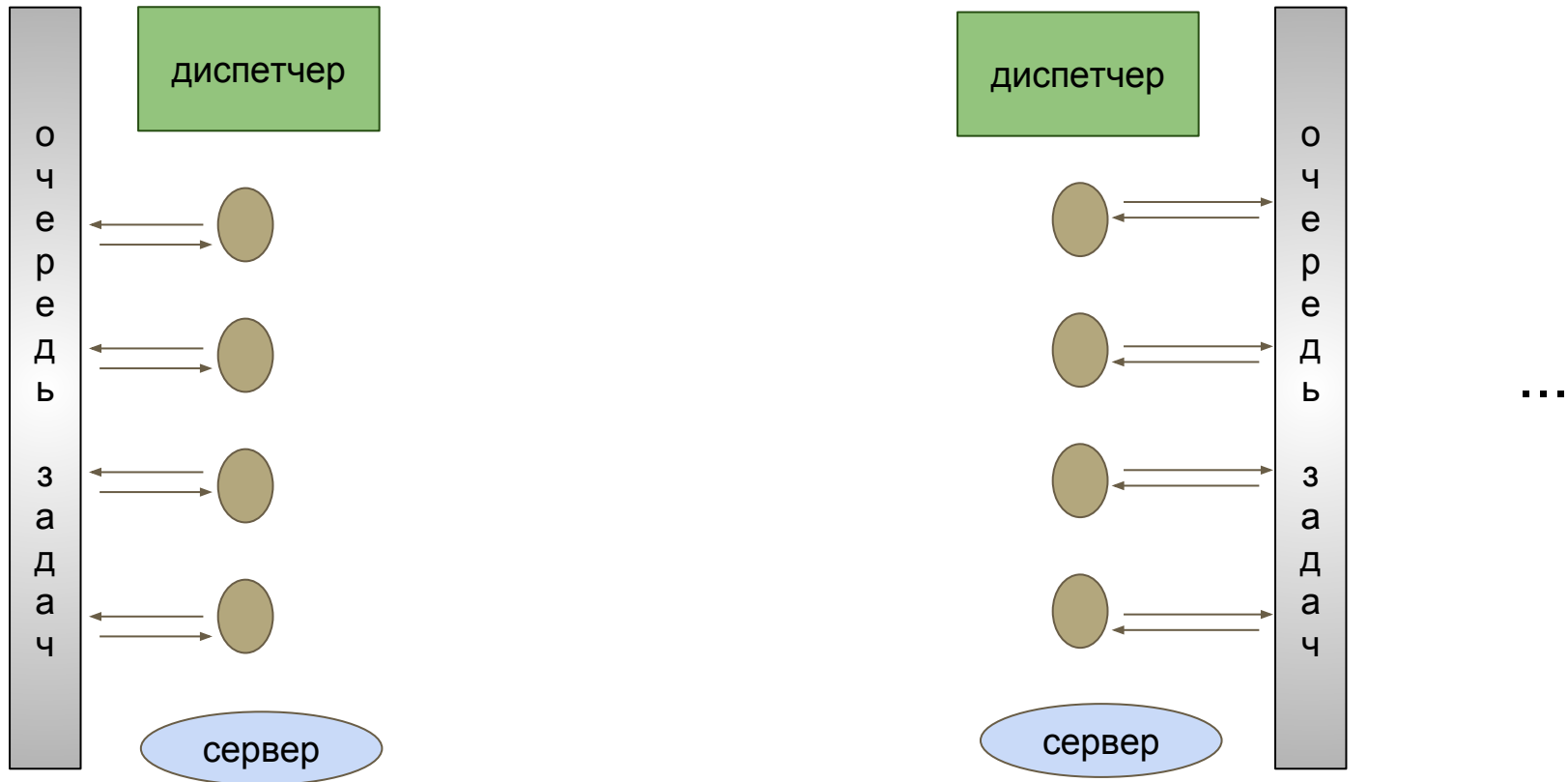
Преимущества: более широкий класс решаемых задач, автоматическое перераспределение ресурсов, большая часть программной реализации скрыта от пользователя.

Недостатки: сложность программной реализации, накладные расходы на управление распределением.

Схема взаимодействия процессов при распределении задач

1 процесс

2 процесс

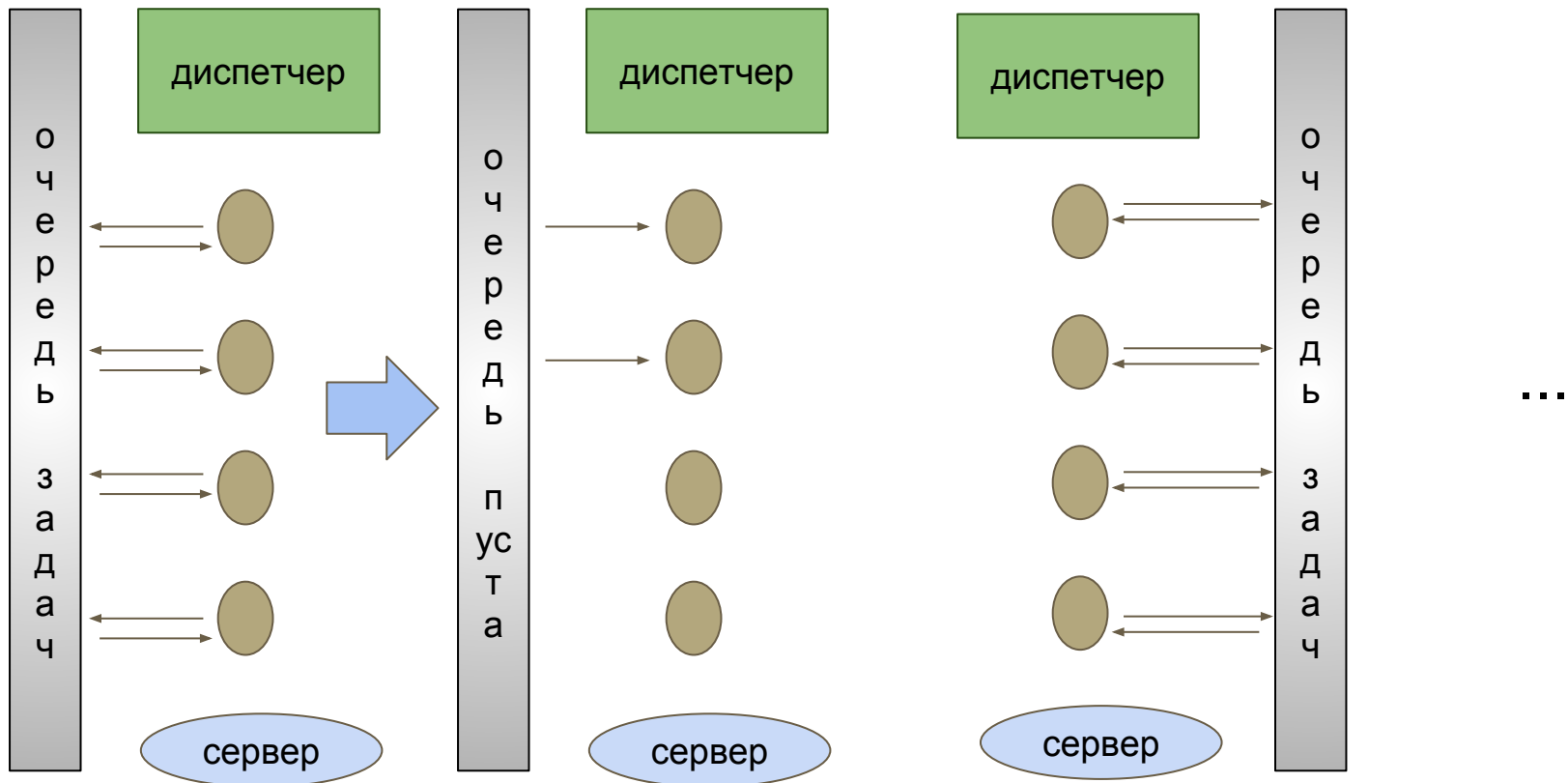


● - потоки, выполняющие подзадачи из очереди задач

Схема взаимодействия процессов при распределении задач

1 процесс

2 процесс




 - потоки, выполняющие подзадачи из очереди задач

Схема взаимодействия процессов при распределении задач

1 процесс

2 процесс

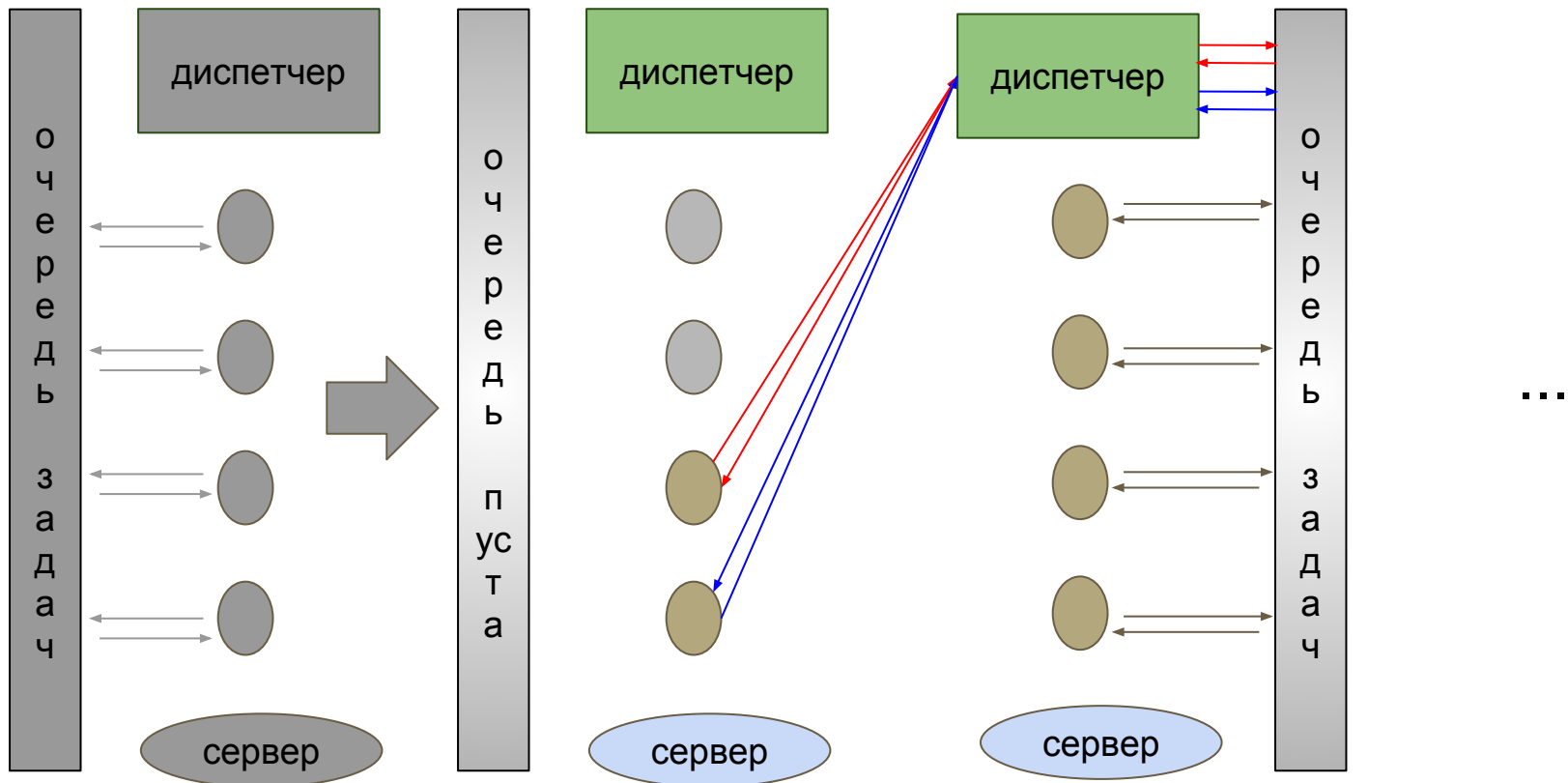


Схема взаимодействия процессов при динамическом подключении процессов

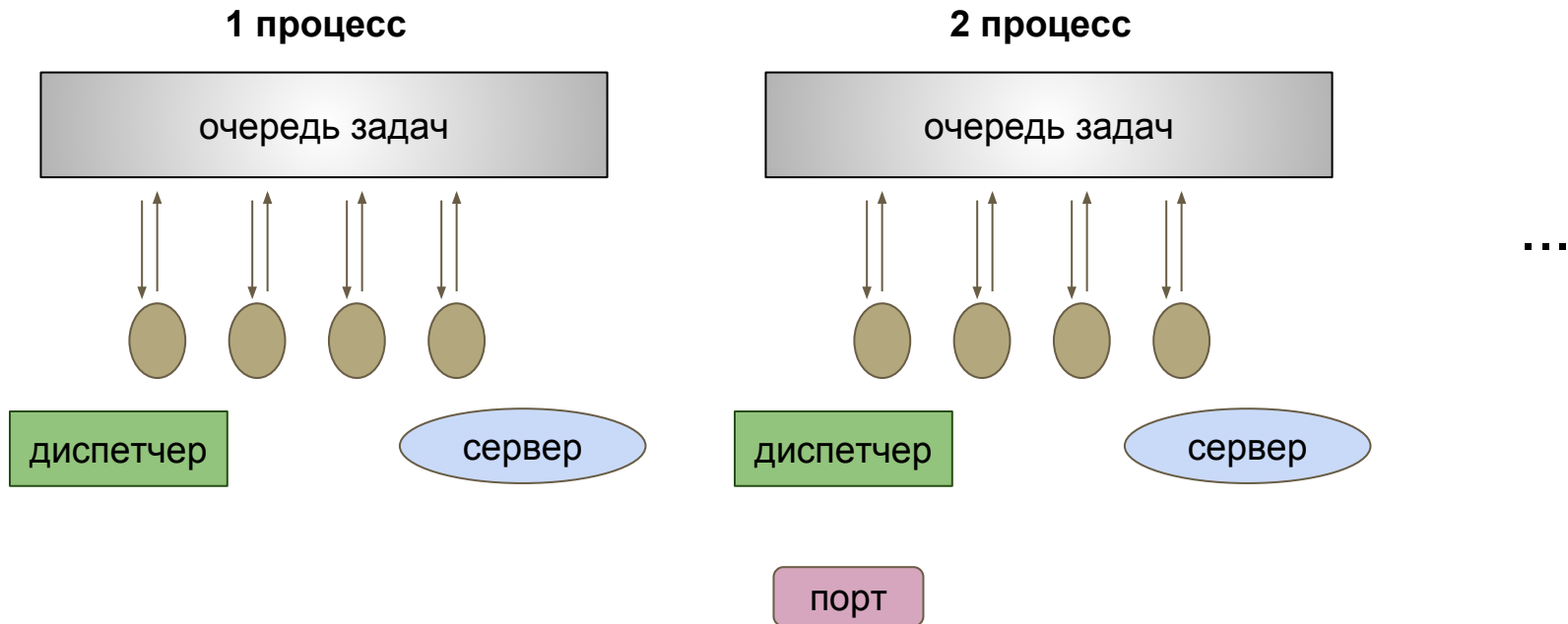


Схема взаимодействия процессов при динамическом подключении процессов

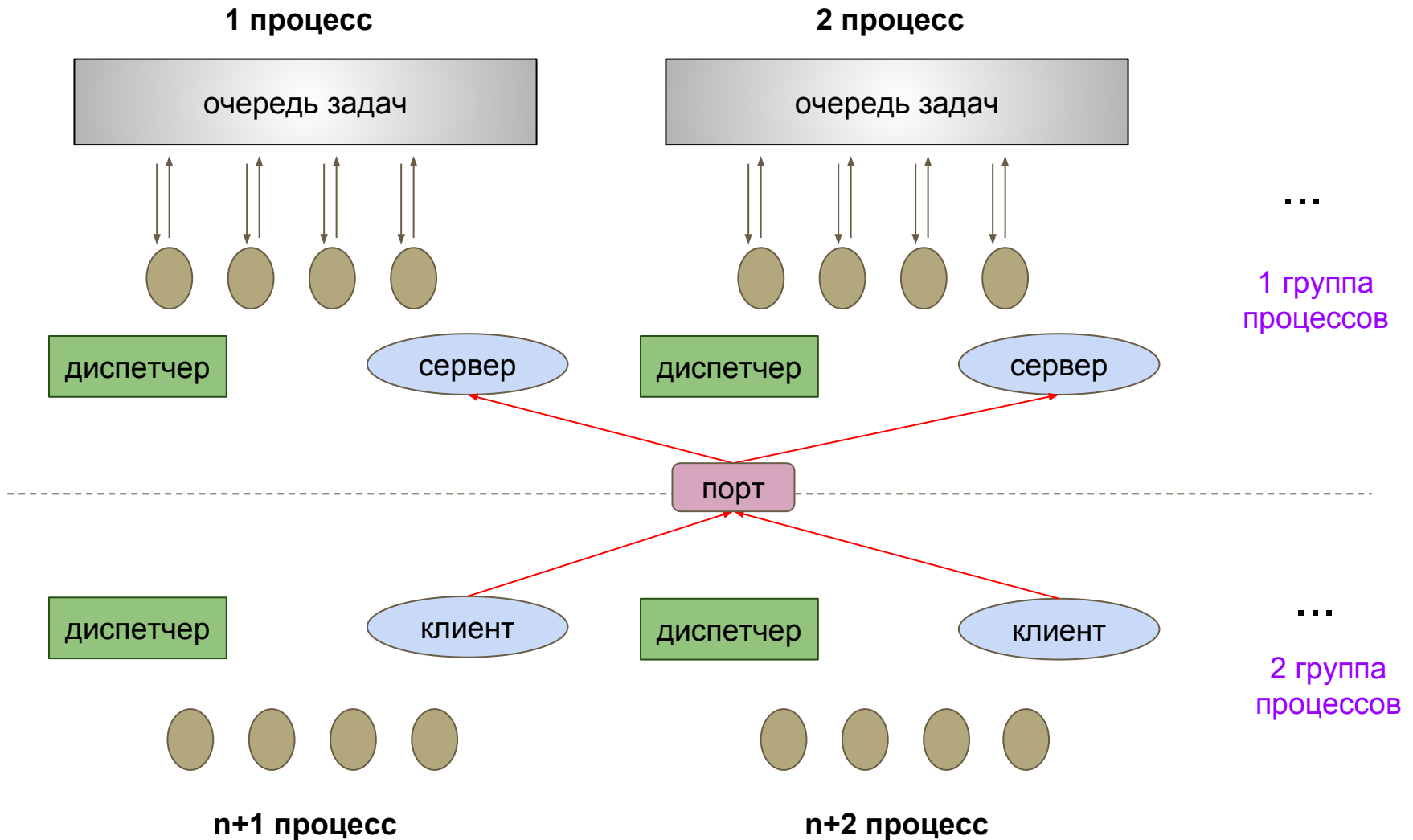
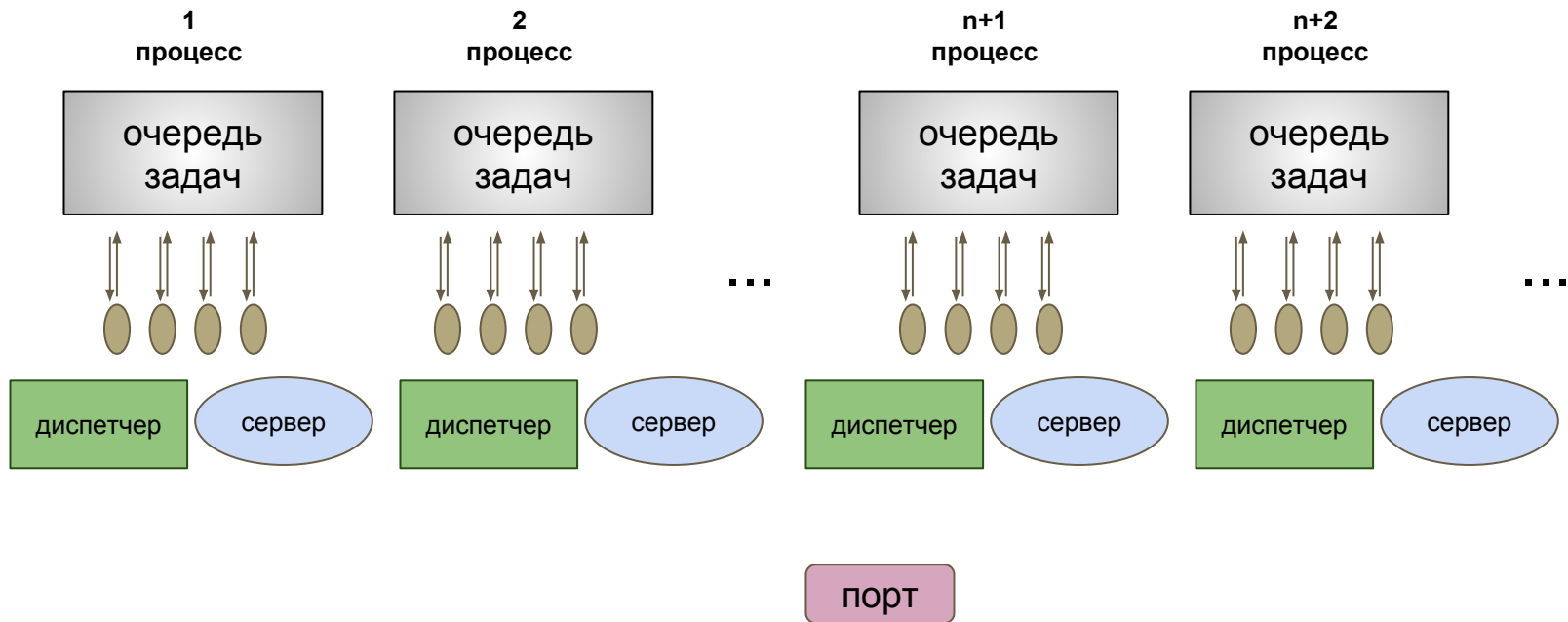


Схема взаимодействия процессов при динамическом подключении процессов



Программные средства и архитектура

- Программные средства:
 - POSIX Threads, MPI
- Архитектура библиотеки:
 - многопоточное приложение;
 - узел MPI - вычислительный узел компьютера;
 - число потоков равно числу ядер;
 - дополнительно присутствуют потоки управления (диспетчер и сервер/клиент).

Заключение

- Был разработан прототип библиотеки с возможностями
 - динамического распределения вычислений между потоками и процессами;
 - динамического подключения новых процессов к решению подзадач;
- Были изучены основы параллельного программирования на MPI и POSIX Threads

Дальнейшая работа

Требуется

- организовать корректные связи между сервером и клиентом при их объединении;
- продумать более эффективное распределение вычислений между узлами;
- включить возможность зависимости подзадач.

Спасибо за внимание!

