Программная архитектура системы передачи интенсивного потока данных в распределенных системах

Автор: **Щапов В.А.** (1, 2)
1) ИМСС УрО РАН
2) ПНИПУ

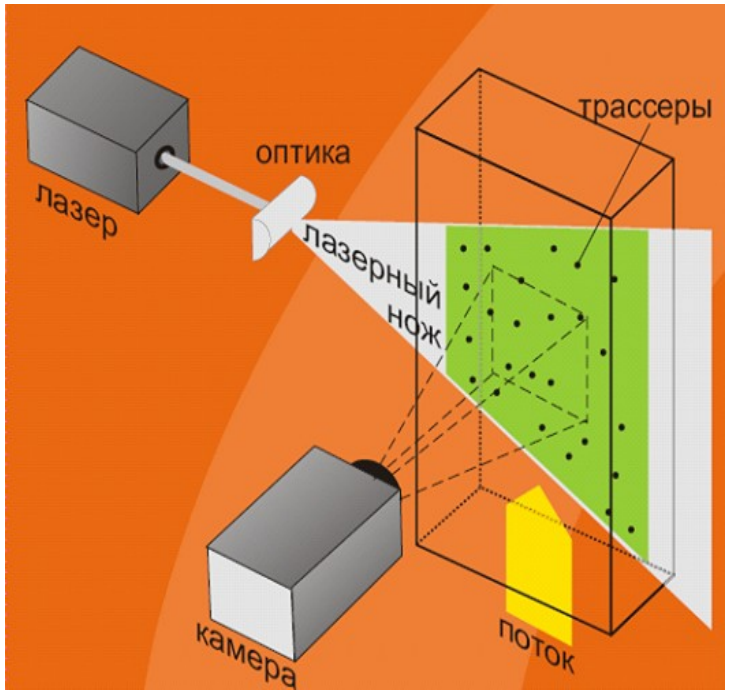
Проекты

- «Инициатива GIGA UrB RAS»
  - Создание высокоскоростной научно-образовательной оптической магистрали УрО РАН «Архангельск-Екатеринбург»
  - Доступная к использованию в 2012 году структура DWDM тракта на участке «Пермь-Екатеринбург»: 3λ * 10 Гбит/с

- «Распределенный PIV»
  - Распределенная обработка потока экспериментальных данных в реальном времени на удаленных суперкомпьютерах

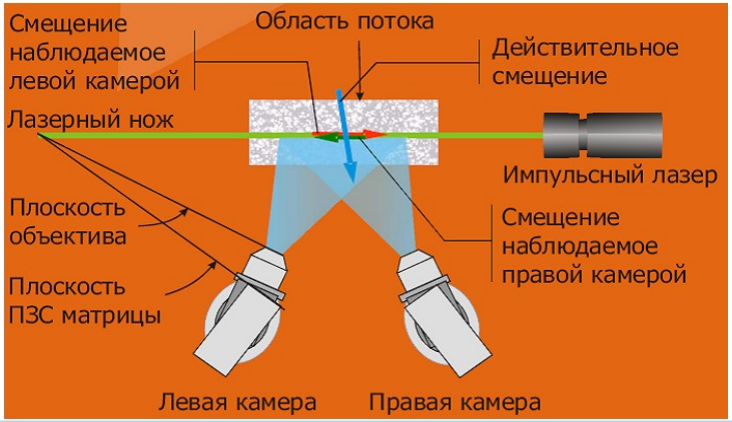
Эксперимент PIV

- **Метод PIV (Particle Image Velocimetry)** – оптический метод измерения полей скорости жидкости или газа в выбранном сечении потока
- Вычислительная ограниченность сдерживает развитие математического аппарата и возможности эксперимента
- Перенос вычислений на многопроцессорные системы позволит:
  - использовать ресурсоемкие, но высокоточные алгоритмы,
  - избегать хранения гигантских объемов избыточной информации,
  - обрабатывать измерения в реальном времени,
  - проводить эксперименты с обратной связью.



Метод позволяет использовать параллельную обработку потока данных

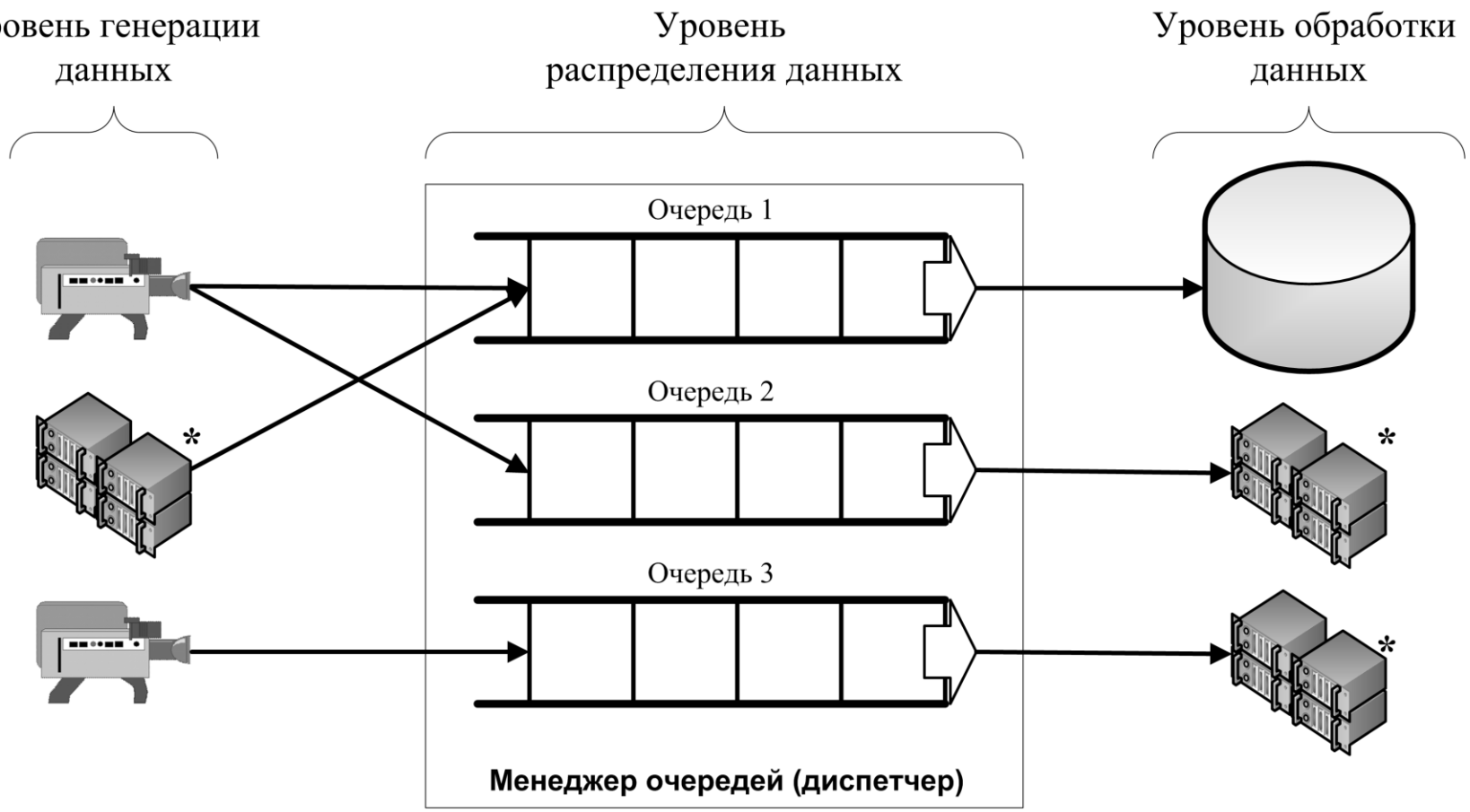
Поток данных можно разделить на блоки (измерения), допускающие независимую обработку на вычислителях



Параллельная обработка потока данных online

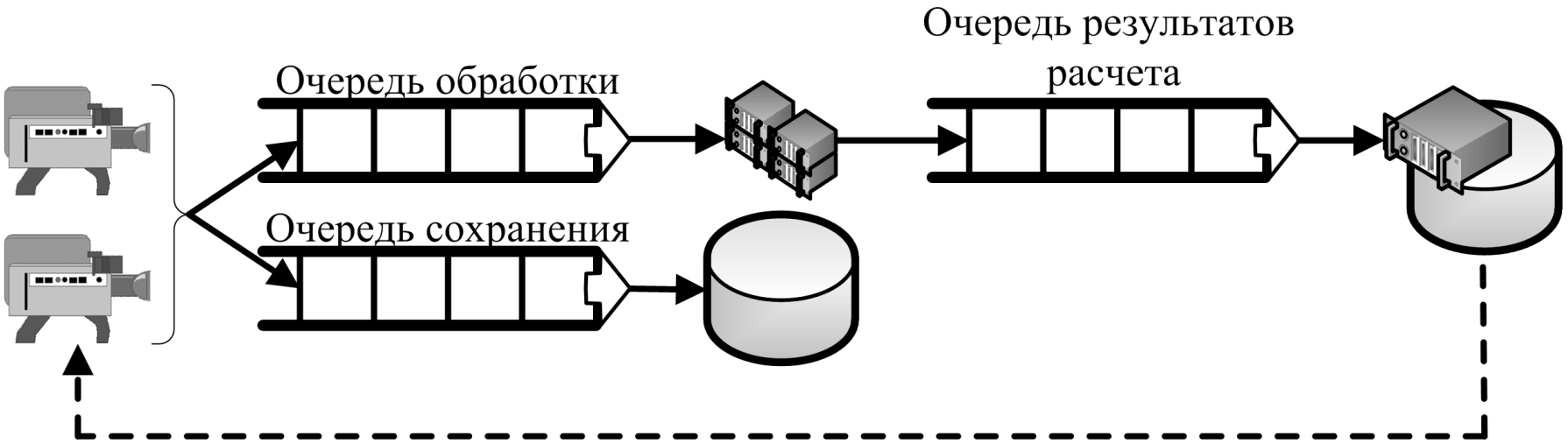
- Исходный поток представим в виде сообщений, допускающих независимую обработку.
- Сообщения распределяются по запросам в порядке очереди. Нет однозначного отображения сообщений на вычислительные узлы:
  - не нужна синхронизация между вычислительными узлами в кластере;
  - возможно изменять число задействованных вычислительных узлов во время расчета;
  - минимизация объема потерянной информации при выходе из строя вычислительного узла.

Трехуровневая архитектура системы обработки данных

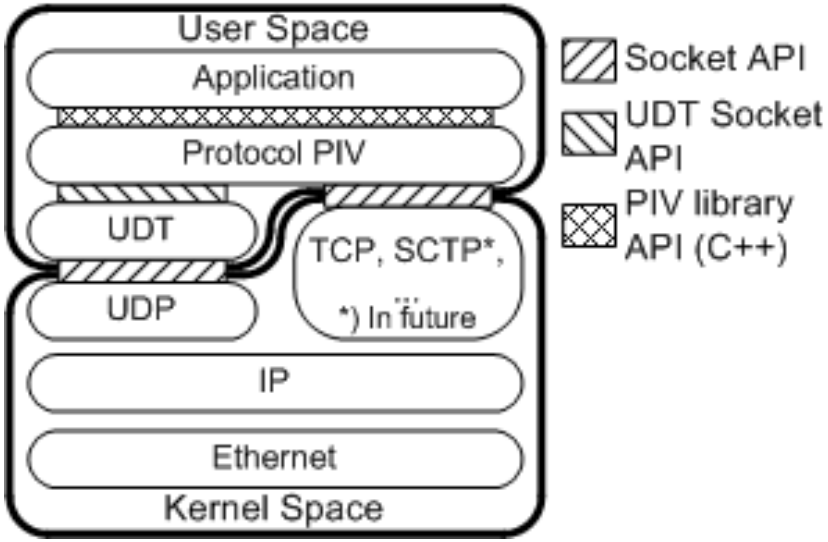


*) Одно устройство может находиться и на уровне генерации данных, и на уровне обработки данных

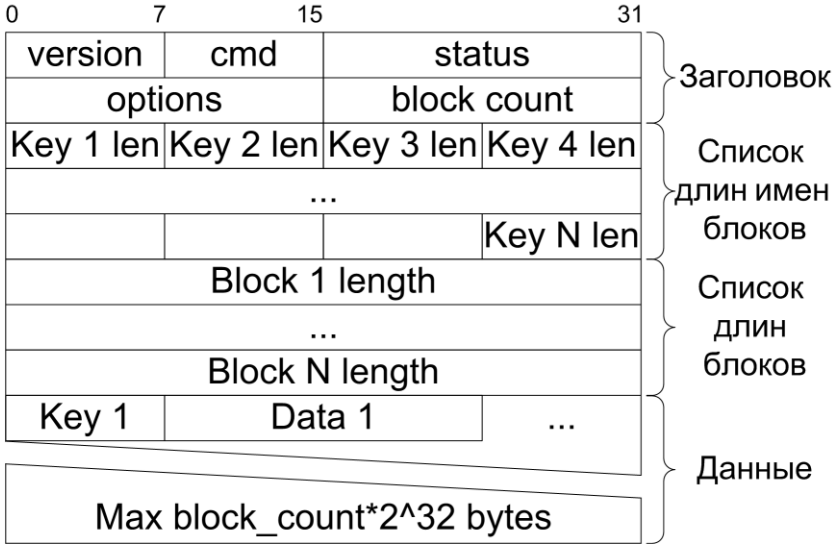
Схема взаимодействия компонент системы



Протокол PIV



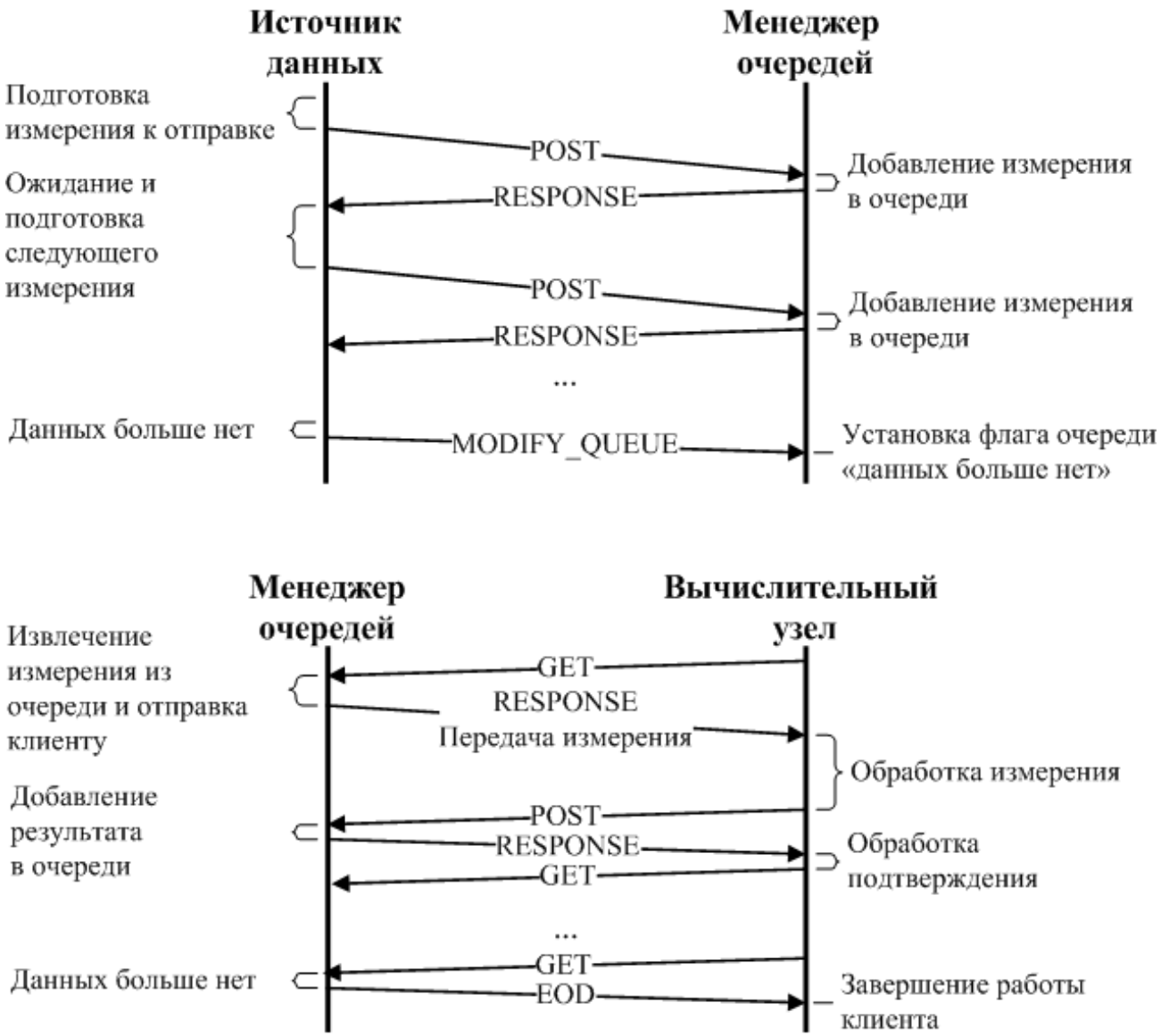
Положение в стеке технологий ОС



Формат пакета

- RPC-подобный протокол;
- до 65535 блоков данных в одном сообщении;
- до 4 Гб данных в одном блоке данных;
- текущая реализация использует TCP и UDT.

Временная диаграмма



Уровень генерации данных

- Загрузка и отправка исходных данных из источника на уровень распределения.
- Взаимодействие с сервером очередей через клиентскую библиотеку.

- В PIV-расчетах – реализовано приложение ввода/вывода для получения данных от ActualFlow с последующей передачей на сервер очередей и сохранения результатов, полученных от сервера в файлах.

Уровень обработки данных

- Расчет исходных данных с применением каких-либо алгоритмов.
- Сохранение данных в хранилищах.
- Передача данных в сторонние системы.
- …

- В PIV-расчетах – реализован фреймворк для упрощения написания расчетных приложений для суперкомпьютера.

Уровень распределения данных

- Передача данных между уровнями генерации и обработки данных.
- Распределение очередей данных по вычислителям на уровне обработки.
- Обработка большого количества TCP/UDT соединений от источников данных и вычислителей.

- Стратегия распределения данных по узлам – FIFO по запросу.

Программная архитектура

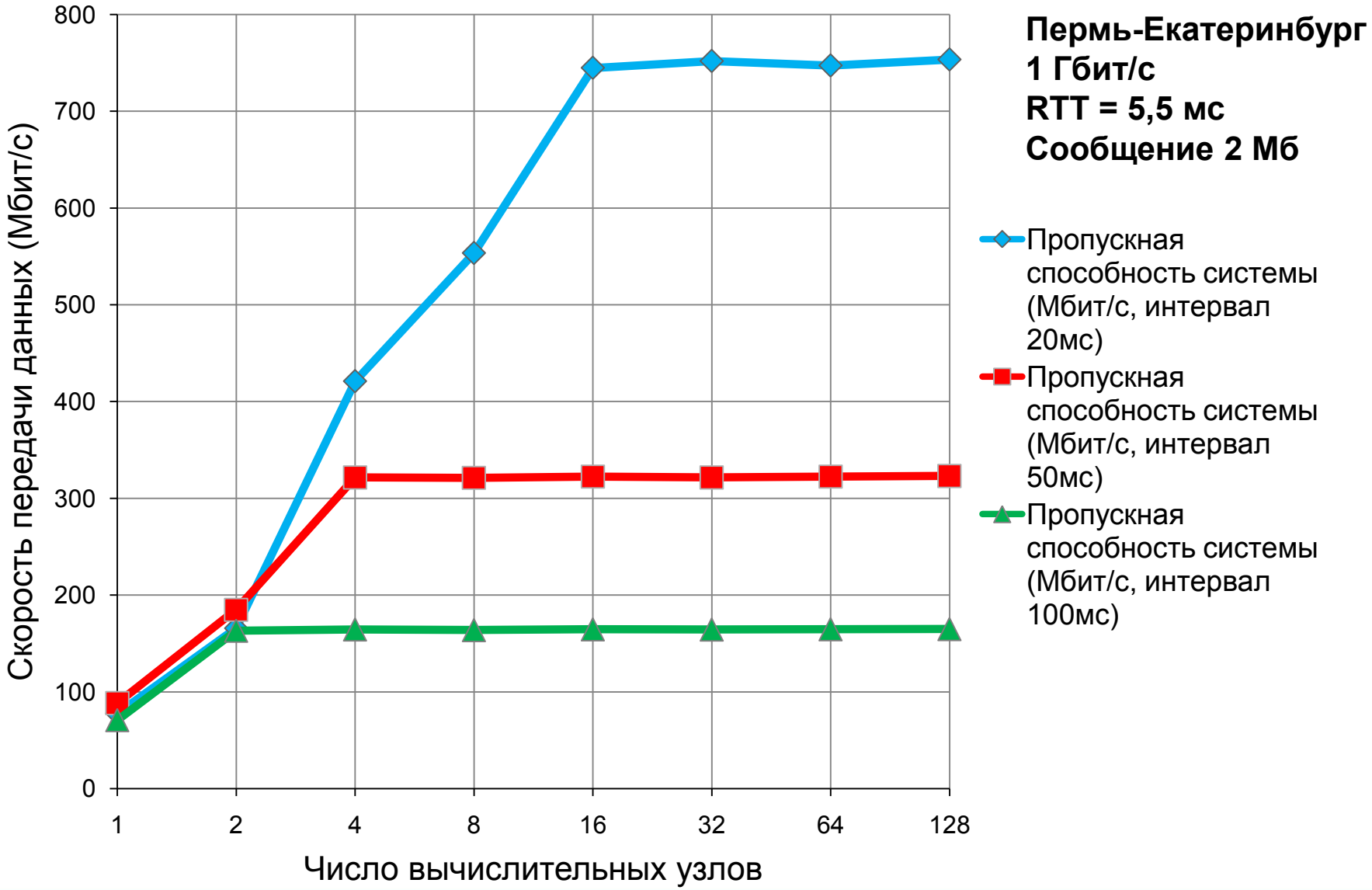
- Кроссплатформенность:
  - Red Hat Enterprise Linux >= 5.
  - SUSE >= 11.
  - Windows XP, 7 (только 32bit).

- Компиляторы:
  - GCC.
  - Intel C++.
  - Microsoft Visual C++ 2010 (10.0).

Программная архитектура

- Технологии:
  - C++.
  - Boost C++ Libraries (http://www.boost.org/).
- Асинхронный, event driven I/O сервера очередей (Boost.Asio):
  - Время передачи данных >> времени CPU.
  - 4 потока AIO ≈ 60 потоков SyncIO.
  - Сокращение числа переключения контекста, уменьшение нагрузки на сервер, возможность обрабатывать больший поток данных.

Оценка эффективности



Заключение

- Предложена программная архитектура системы передачи интенсивного потока данных в распределенных системах.
- Сформулированы ограничения на класс прикладных задач, для которых возможно применение разработанного подхода.
- Спроектирован протокол, алгоритм диспетчеризации данных и разработано программное обеспечение.
- Планируемые направления работ:
  - Разработка методики оценки требуемых вычислительных и коммуникационных ресурсов;
  - Исследование и улучшение алгоритма распределения очереди исходных данных;
  - Исследование, анализ характеристик и областей применения транспортных протоколов и технологий внутренних коммуникаций суперкомпьютеров.

Спасибо за внимание!

Автор: **Щапов В.А.** (1, 2)

shchapov@icmm.ru

1) ИМСС УрО РАН
2) ПНИПУ