

Учебный курс

Основы операционных систем

Лекция 13

кандидат физико-математических наук, доцент

Карпов Владимир Ефимович

Планирование запросов

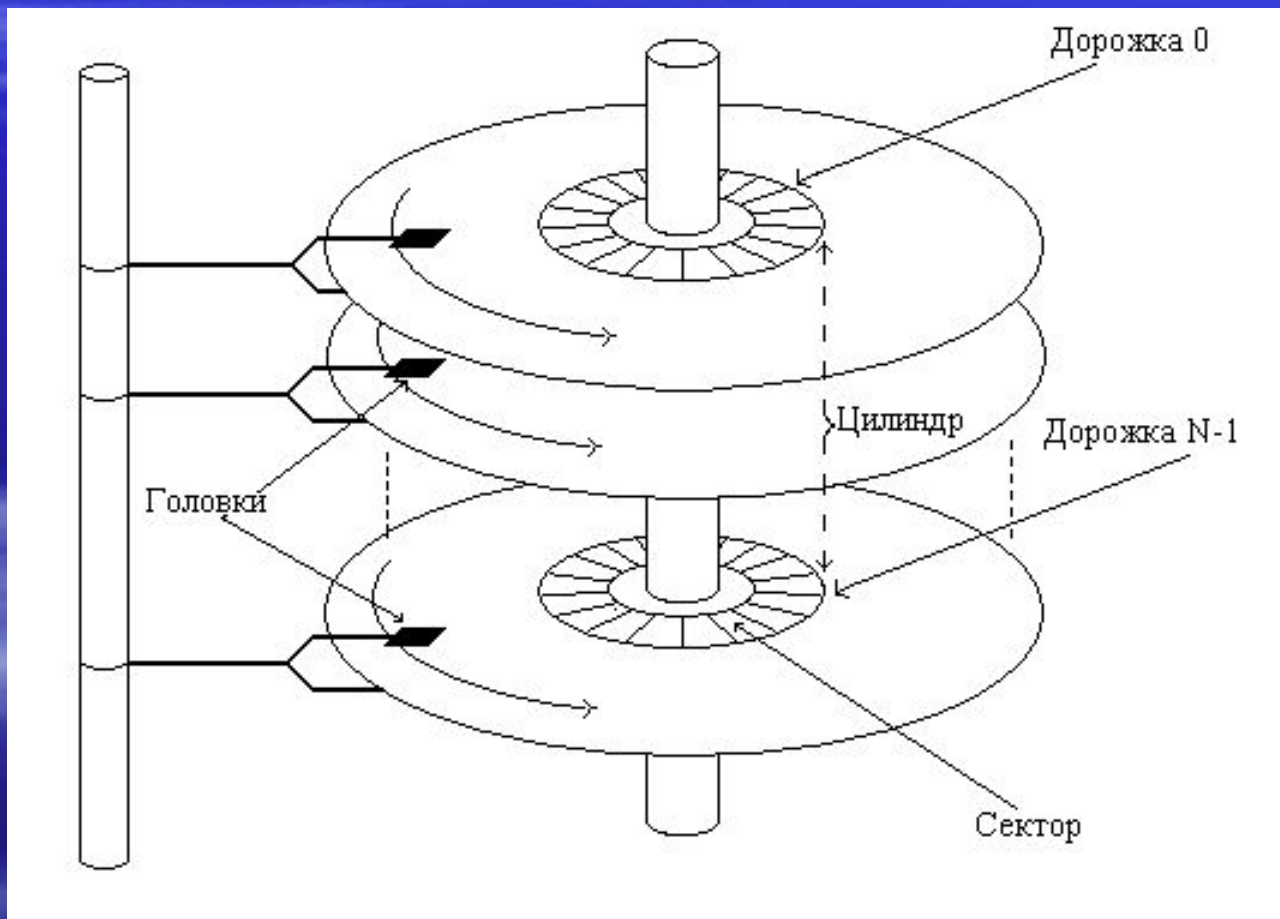
Для блокирующихся и асинхронных системных
ВЫЗОВОВ

- При занятости устройства запрос ставится в очередь к данному устройству.
- После освобождения устройства необходимо принять решение: какой из запросов в очереди инициировать следующим – планирование запросов.

Действия по планированию запросов могут быть частично или полностью делегированы драйверу устройства – функция `strategy` в интерфейсе драйвера

Алгоритмы планирования запросов к жесткому диску

Строение жесткого диска



Алгоритмы планирования запросов к жесткому диску

Параметры планирования

- Запрос полностью характеризуется:
 - типом операции
 - номером цилиндра
 - номером дорожки
 - номером сектора
- Параметр планирование – время, необходимое для выполнения запроса.
 - Время выполнения запроса = transfer time + positioning time
 - Positioning time = seek time + positioning latency

Единственным параметром запроса остается seek time – время пропорциональное разнице между номером цилиндра в запросе и номером текущего цилиндра

Алгоритмы планирования запросов к жесткому диску

Диск имеет 100 цилиндров (от 0 до 99)
Очередь запросов: 23, 67, 55, 14, 31, 7, 84, 10
Текущий цилиндр – 63

- Алгоритм FCFS (First Come First Served)

63 -> 23 -> 67 -> 55 -> 14 -> 31 -> 07 -> 84 -> 10
Всего перемещение на 329 цилиндров

- Алгоритм SSTF (Short Seek Time First)

63 -> 67 -> 55 -> 31 -> 23 -> 14 -> 10 -> 07 -> 84
Всего перемещение на 141 цилиндр

Алгоритмы планирования запросов к жесткому диску

Диск имеет 100 цилиндров (от 0 до 99)
Очередь запросов: 23, 67, 55, 14, 31, 7, 84, 10
Текущий цилиндр – 63

- Алгоритм SCAN

63 -> 55 -> 31 -> 23 -> 14 -> 10 -> 07 -> 0 -> 67 -> 84

Всего перемещение на 147 цилиндров

- Алгоритм LOOK

63 -> 55 -> 31 -> 23 -> 14 -> 10 -> 07 -> 67 -> 84

Всего перемещение на 133 цилиндра

- Алгоритм C-SCAN

63 -> 55 -> 31 -> 23 -> 14 -> 10 -> 07 -> 0 -> 99 -> 84 -> 67

- Алгоритм C-LOOK

63 -> 55 -> 31 -> 23 -> 14 -> 10 -> 07 -> 84 -> 67

Часть VI.
Сети и сетевые
операционные системы

Для чего компьютеры объединяют в сети:

- Совместное использование ресурсов (как физических, так и информационных)
- Ускорение вычислений
- Повышение надежности работы
- Общение пользователей

Сетевые и распределенные ОС

■ Сетевые ОС

- Пользователь должен знать, что компьютер находится в сети, где находятся удаленные ресурсы и как до них добраться
- От автономной операционной системы отличается наличием дополнительной сетевой части

■ Распределенные ОС

- Пользователь работает как в автономной операционной системе
- Операционная система полностью отличается от автономных ОС и функционирует на всех компьютерах сети

Отличия взаимодействия удаленных и локальных процессов

- В основе взаимодействия локальных процессов лежит использование разделяемой памяти, а в основе взаимодействия удаленных процессов – передача сообщений
- Информация между удаленными процессами зачастую передается через процессы-посредники, обитающие на компьютерах, отличных от компьютеров получателя и отправителя
- Удаленное взаимодействие должно строиться исходя из первоначального предположения о ненадежности связи
- Адреса взаимодействующих процессов должны быть уникальны в рамках всей сети
- Необходимо обеспечить взаимоисключения процессов при обращении к линии связи между компьютерами

Основные вопросы логической организации связи между удаленными процессами

- Как прокладывать физические линии связи?
- Как исключить возникновение race condition при их использовании?
- Какие виды интерфейсов могут быть предоставлены пользователю?
- Как организована адресация удаленных процессов?
- Как информация передается от отправителя к получателю через узлы посредники?

Понятие протокола

Для обмена почтовыми сообщениями между людьми соответствующие службы связи должны договориться о:

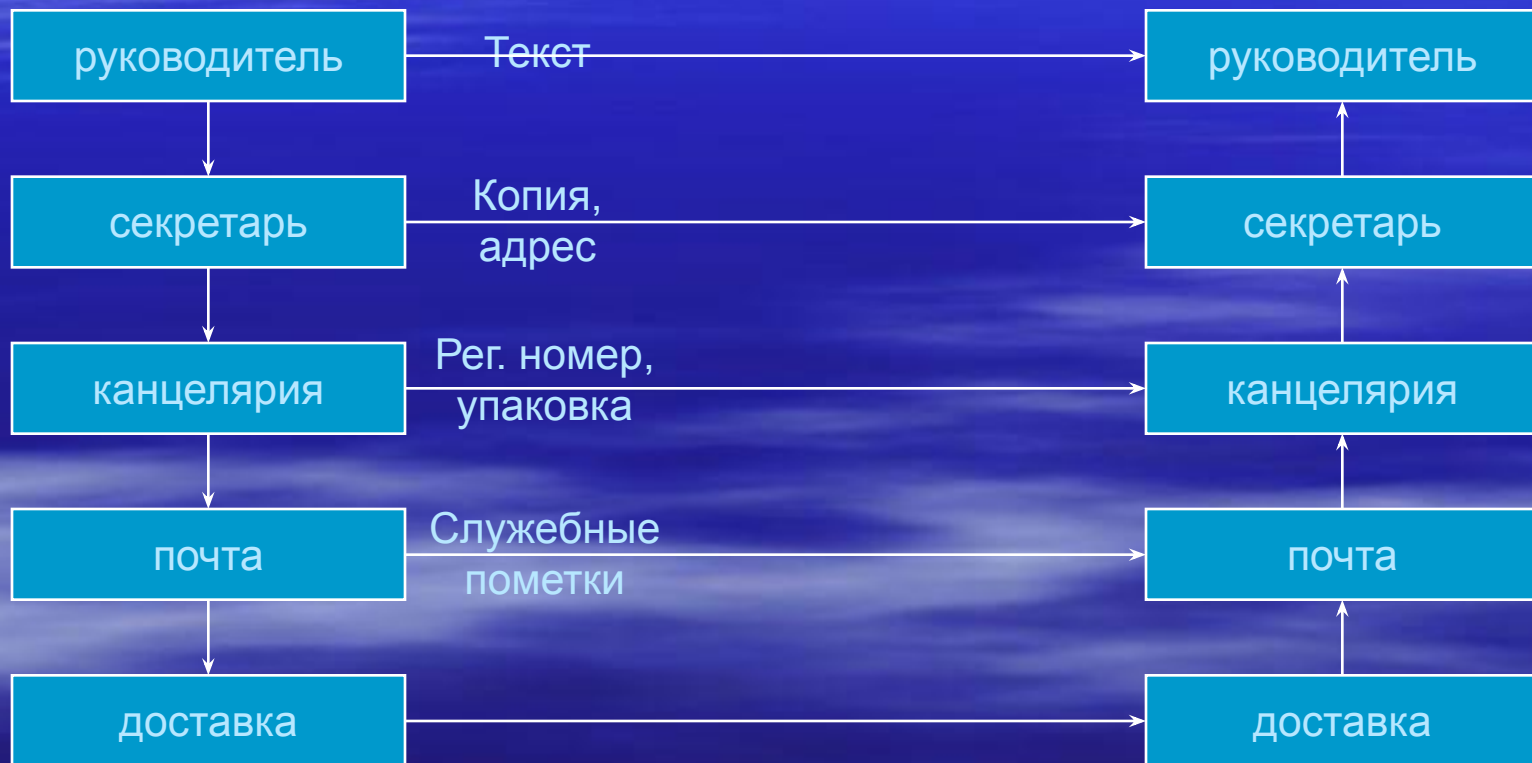
- Виде почтовых сообщений
- Содержанию служебной информации
- Формате представления служебной информации

Для взаимодействия удаленных процессов сетевые части операционных систем также должны руководствоваться определенными соглашениями (поддерживать определенные протоколы)

Многоуровневая модель построения сетевых вычислительных систем



Многоуровневая модель построения сетевых вычислительных систем

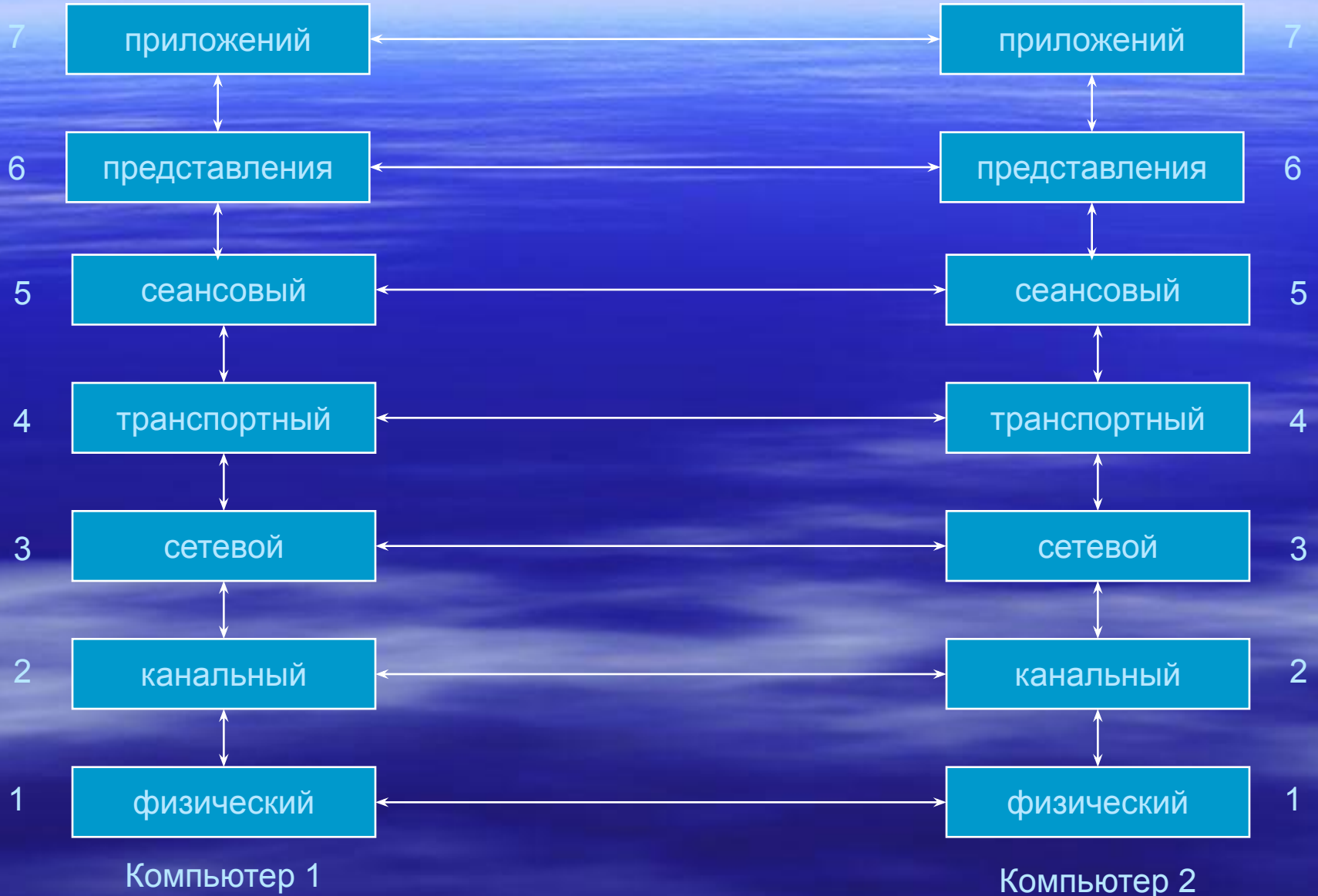


Многоуровневая модель построения сетевых вычислительных систем

Формальный перечень правил, определяющих последовательность и формат сообщений, которыми обмениваются сетевые компоненты вычислительных систем, лежащие на одном уровне называется **сетевым протоколом**

Вся совокупность вертикальных и горизонтальных протоколов (интерфейсов и сетевых протоколов) называется **семейством протоколов** или **стеком протоколов**

Модель OSI/ISO



Одноуровневая адресация

- Процесс запрашивает свою ОС о возможности использовать адрес
- ОС опрашивает все другие ОС о возможности присвоения адреса
- После сбора всех ответов ОС принимает решение
- ОС извещает процесс о принятом решении

Двухуровневая адресация: удаленные адреса

- Проблема уникальности адресов
- Проблема разрешения адресов

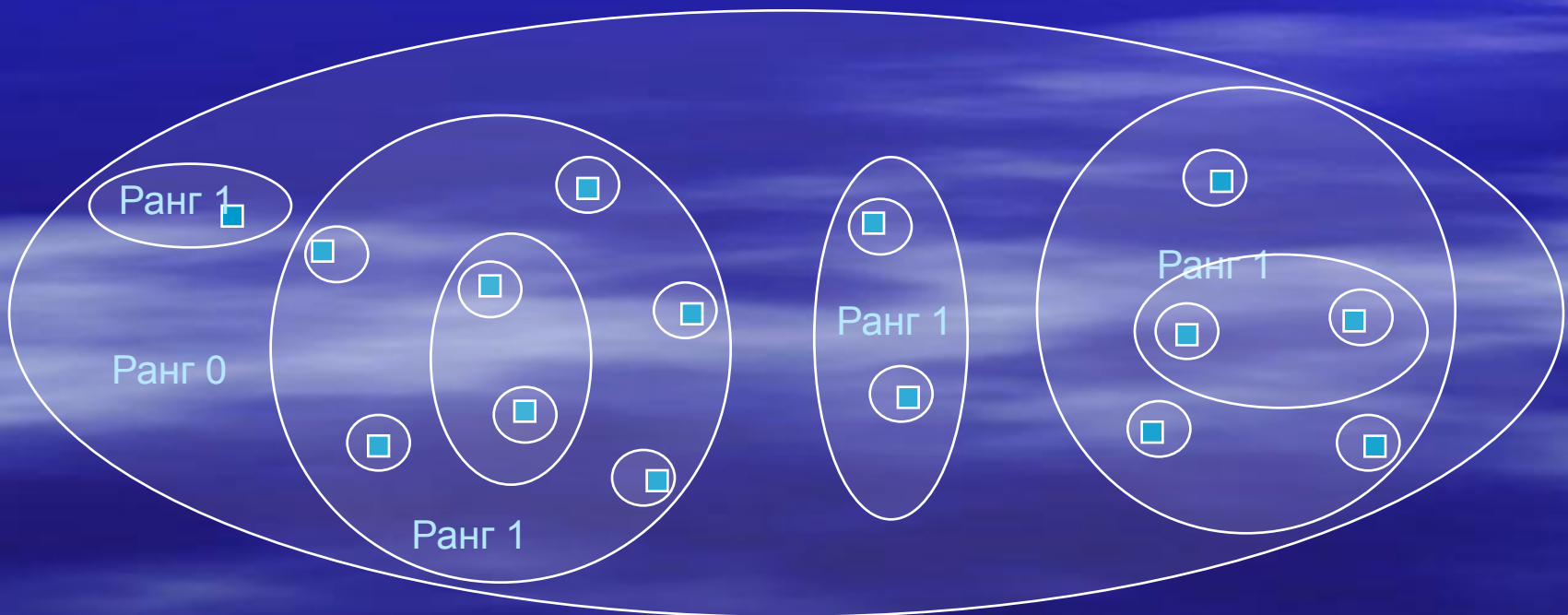
Централизованный подход

- Информация о всех символьных и числовых адресах хранится в специальном файле
- Изменения вносятся в файл только администратором сети на специальном компьютере
- Файл с изменениями рассылается на все компьютеры сети

Двухуровневая адресация: удаленные адреса

- Проблема уникальности адресов
- Проблема разрешения адресов

Распределенный подход (на примере DNS)



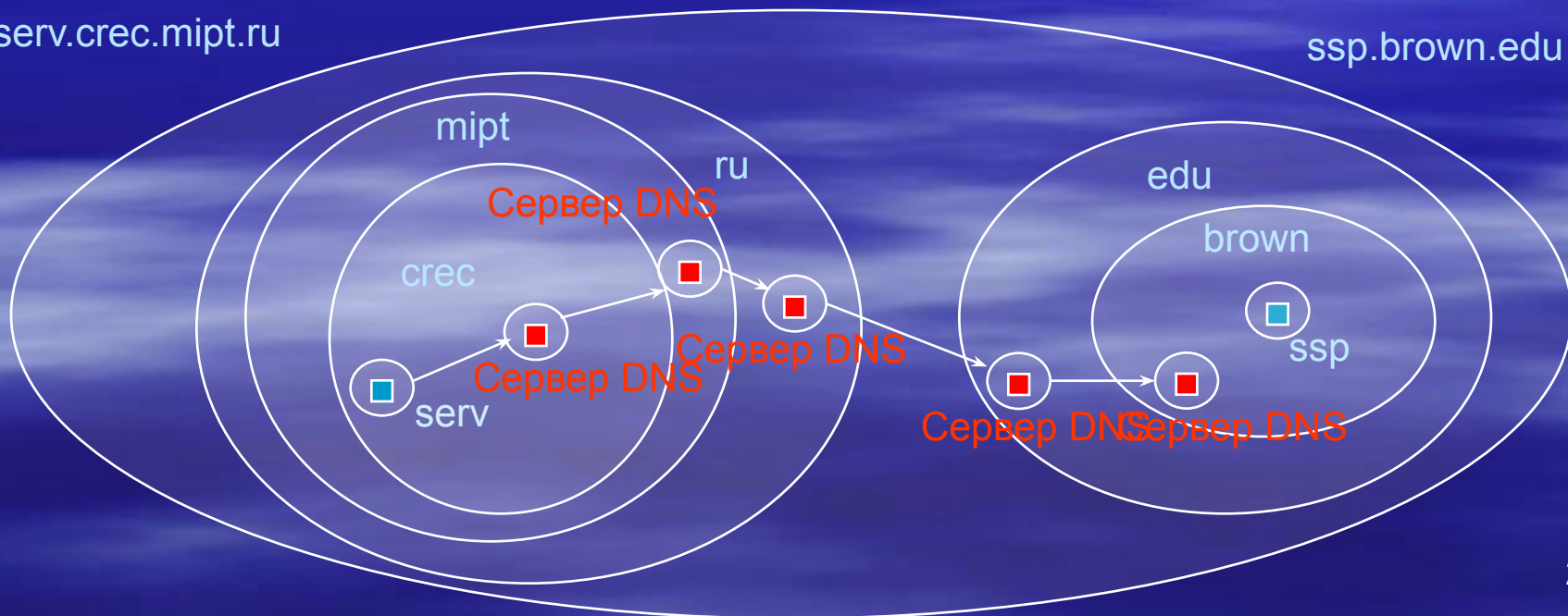
Двухуровневая адресация: удаленные адреса

- Проблема уникальности адресов
- Проблема разрешения адресов

Распределенный подход (на примере DNS)

serv.crec.mipt.ru

ssp.brown.edu



Двухуровневая адресация: локальные и полные адреса

Локальные адреса

- PID не подходит в качестве локального адреса
- Для локальной адресации создается специальное адресное пространство – пространство **сетевых портов**
- Для разных средств связи – разные пространства портов

Полные адреса

- Полный адрес – <числовой удаленный адрес, порт>
- Полный адрес получил название socket (сокет)

Проблемы маршрутизации в сети

- Маршрутизация от источника передачи данных
- Одношаговая маршрутизация

Адресат назначения	Адрес очередного компонента сети	Адрес исходящей линии связи
5	20	21
1-4	28	22
default	24	23

Проблемы маршрутизации в сети

- Маршрутизация от источника передачи данных
- Одношаговая маршрутизация
 - Алгоритмы фиксированной маршрутизации
 - Алгоритмы простой маршрутизации
 - Случайная
 - Лавинная
 - Маршрутизация по прецедентам
 - Алгоритмы динамической маршрутизации
 - Дистанционно-векторные алгоритмы
 - Алгоритмы состояния связей

Интерфейсы сетевых средств связи

- Протоколы без установления логического соединения или протоколы обмена датаграммами (datagramm) - connectionless
- Протоколы, требующие установления логического соединения – connection-oriented