9.09.2024

Сетевые средства:

1. Серверная часть ОС
2. Средства запроса доступа к удалённым ресурсам – клиентская часть ОС
3. Транспортное средство ОС, которое с коммуникационной системой обеспечивает передачу сообщений между компьютерами сети

(Сервер может быть клиентом, а клиент сервером)

Локальные сети можно подразделять в зависимости от используемой сетевой и операционной системы на: серверные, одноранговые, комбинированные.

(Серверные – клиент-серверные сети  
Одноранговые – все равны

Комбинированные – есть и те и те)

Сетевые службы и приложения

Предоставление пользователям совместного доступа к определённому типу ресурсов, например, к файлам – называют также предоставлением сервиса. Обычно сетевая операционная система поддерживает несколько видов сетевых сервисов: файловая, сервис печати, удалённого доступа и т. д.

Программы, реализующие сетевые сервисы, относятся к классу распределённых программ. Распределенная программа – это программа, которая состоит из нескольких взаимодействующих частей, причём каждая часть, как правило, выполняется на отдельном компьютере сети. Это системные распределенные программы, реализующие сетевые сервисы. Они представляют собой пару клиент-сервер и являются неотъемлемыми компонентами сетевой ОС.

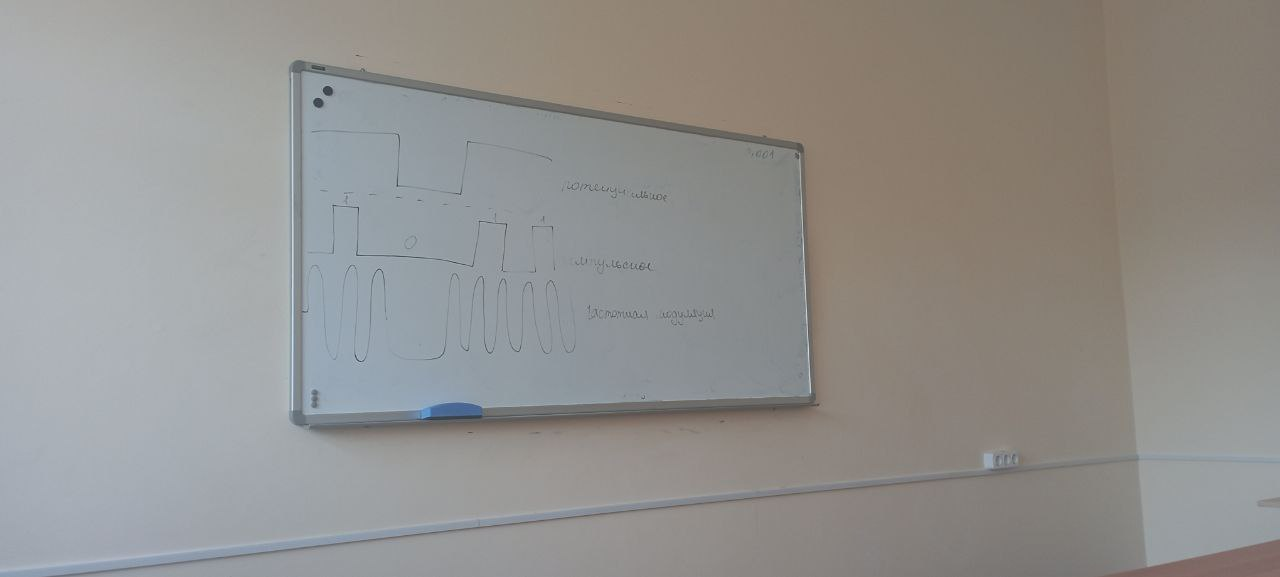
Распределенное приложение выполняет определенную законченную работу по выполнению прикладной задачи. Например, часть приложения, выполняющаяся на компьютере, может поддерживать графический интерфейс, а другая его часть работать на мощном выделенном компьютере и заниматься обработкой данных. Распределенные приложения в полной мере используют потенциальные возможности распределенной обработки и поэтому часто называются сетевыми приложениями.

Проблемы проектирования распределенных приложений:

1. На сколько частей разбить приложение?
2. Какие функции возложить на каждую часть?
3. Как организовать взаимодействие этих частей так, чтобы в случае сбоев и отказов, оставшиеся части корректно завершали работу?

Физическая передача данных по линиям связи

В вычислительной технике для предоставления данных используется двоичный код. Внутри компьютера нулям и единицам соответствуют дискретные электрические сигналы. Представление данных в виде электрических или оптических сигналов называется кодированием. Существуют различные способы кодирования двоичных цифр. Например, потенциальный способ – при котором единицы соответствуют один уровень напряжения, а нулю – другой. Или импульсный способ, когда для представления цифр используются импульсы различной полярности, либо части импульса – называемые перепад/фронт. Для кодирования данных и передачи их между двумя компьютерами по линиям связи применимы аналогичные подходы. Главное отличие между передачей внутри компа и внешней сети – расстояние, которое приводит к большим искажениям.

В вычислительных сетях применяют как потенциальное, так и импульсное кодирование, а также специфический способ представления данных, который никогда не используется внутри компьютера – модуляция.

При модуляции дискретная информация представляется синусоидальным сигналом той частоты, которую хорошо представляет имеющая линия связи.

Порядок действий при передаче данных по линиям связи:

1. Кодирование
2. Компрессия
3. Преобразование информации из параллельной в последовательную форму (для экономии линии связи)

Элементы, реализующие физическую передачу данных: сетевые адаптеры; сетевые интерфейсы коммутаторов, маршрутизаторов и т. д. и аппаратура для передачи данных (модемы).

Потенциальное или импульсное кодирование применяется на каналах высокого качества, а модуляция предпочтительнее в том случае, когда канал вносит сильное искажение.

Обычно модуляция используется в глобальных сетях при передачи данных через телефонные линии, которые были разработаны для передачи голоса в аналоговой форме.

На способ передачи сигналов влияет количество проводов в линиях связи между компьютерами в линиях связи. Чтобы снизить её стоимость стараются сократить количество проводов. Для этого используют не параллельную передачу всех бит одного байта, а последовательную, побитную.

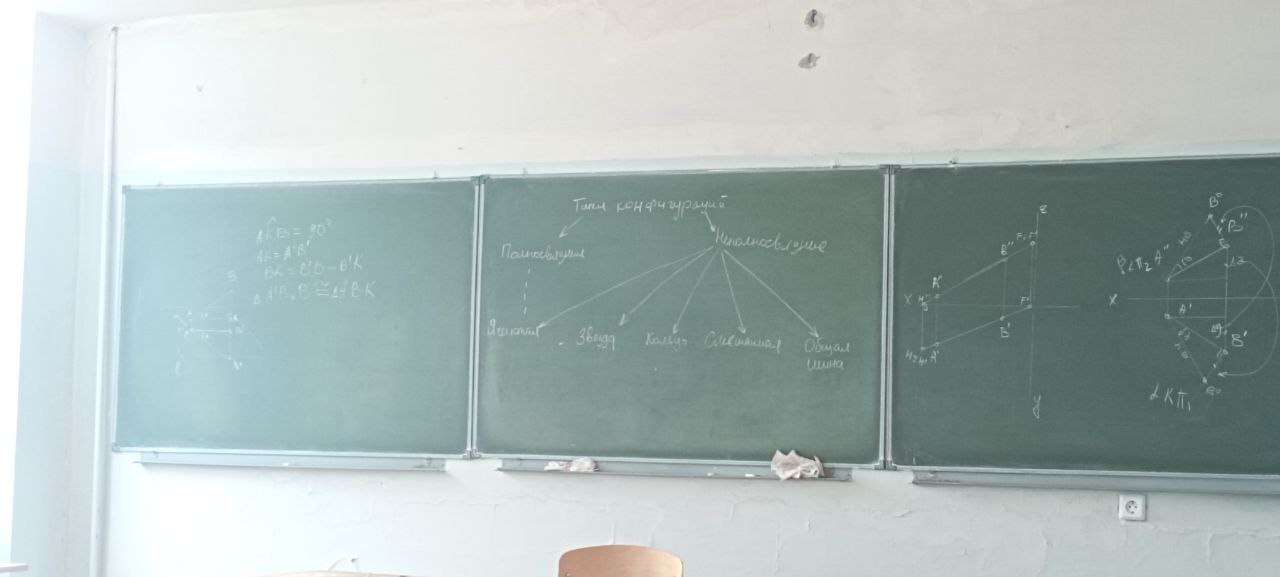
Для более надежной передачи данных часто используется подсчёт контрольной суммы.

20.09.2024

Топология физических связей

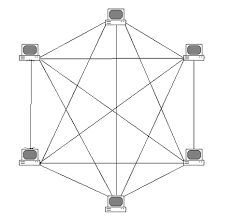
Под топологией сети понимается конфигурация графа, вершинам которого соответствуют конечные сетевые узлы (например компьютеры) и коммуникационное оборудование (например маршрутизаторы) ареограм?, электрические и информационные связи между ними.

Можно соединять каждый компьютер друг с другом или же связывать их последовательно. В таком случае они будут передавать сообщения друг другу транзитом. При этом транзитные узлы должны быть оснащены специальными средствами, позволяющими выполнять эту специфическую посредническую операцию. Транзитный узел может быть: компьютером и специализированным устройством.

Среди множества возможных конфигураций различают полносвязные и неполносвязные. 

Полносвязная топология

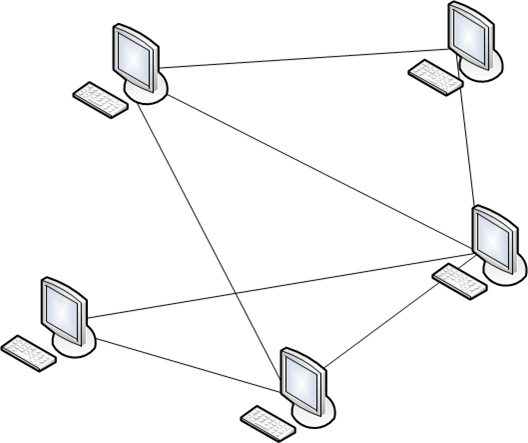
Полносвязная топология соответствует сети, в которой каждый компьютер связан со всеми остальными. Она в крупных сетях применяется редко, т. к. для связи n узлов требуется n(n-1)/2 количество линий связи. Чаще этот вид топологии используется в многомашинных комплексах или в сетях объединияющих небольшое количество компьютеров.



Все другие варианты подключения основаны на неполносвязных топологиях. Когда для обмена данных между двумя компьютерами может потребоваться промежуточная передача (через другие узлы сети).

Ячеистая топология

Ячеистая топология (Mesh) – получается из полносвязной путём удаления некоторых возможных связей. Эта топология допускает соединение большого количества компьютеров и характера для крупных сетей.



Плюсы ячеистой топологии:

1. Высокая отказоустойчивость
2. Повышенная пропускная способность
3. Высокий уровень безопасности

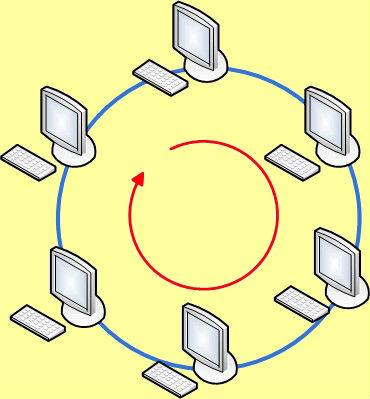
Минусы:

1. Сложность настройки
2. Избыточный расход кабеля
3. Потребность наличия нескольких сетевых интерфейсов

Кольцевая топология

Кольцевая топология – когда устройства соединяются по кругу друг с другом. Данные передаются по кольцу. Главное достоинство – резервирование связей. Кольцо представляет собой удобную конфигурацию для обратной связи – данные, сделав полный оборот, возвращаются к узлу источника.

Свойства кольца часто используются для тестирования связанности сети и поиска узла работающего некорректно. В сетях с кольцевой топологией необходимо принимать специальные меры в случае выхода из строя или отключения какого-либо узла.



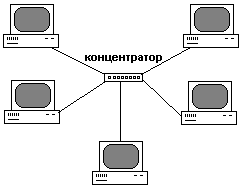
Плюсы:

1. Нет возможности монополизировать сеть (у всех равный доступ)
2. Справедливое совместное использование сети

Минусы:

1. Отказ одного ПК может повлиять на работоспособность сети в целом
2. Добавление или удаление нового узла вынуждает разорвать сеть

Топология звезда



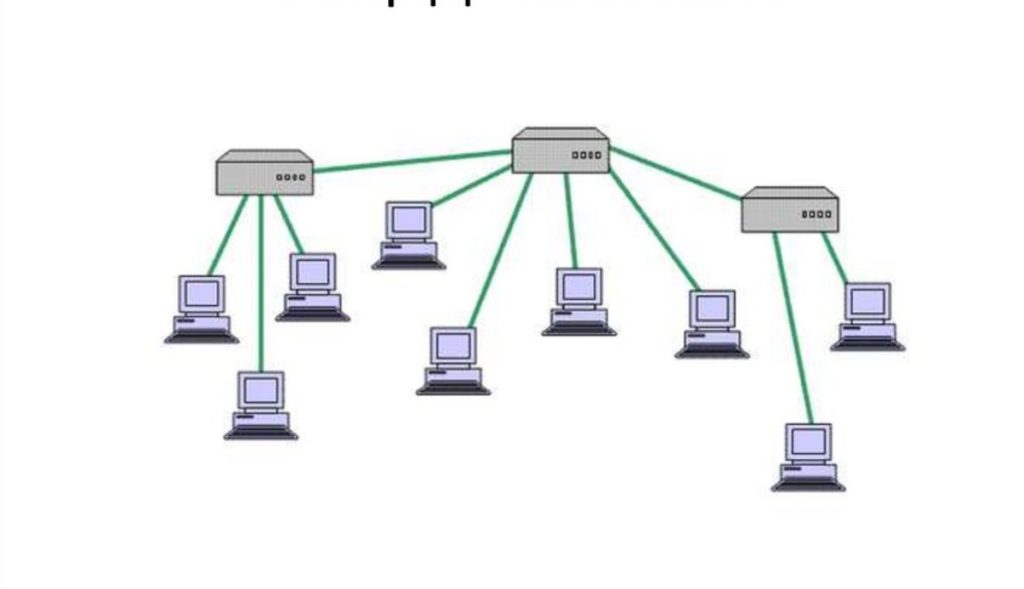
Топология звезда образуется в том случае, когда каждый компьютер с помощью отдельного кабеля подключается к общему центральному устройству, называемого концентратором. Концентратор направляет передаваемую информацию одному или всем компьютерам сети.

В роли концетратора может выступать компьютер, специализированное устройство, многовходовый повторитель, коммутатор, маршутизатор.

Плюсы:

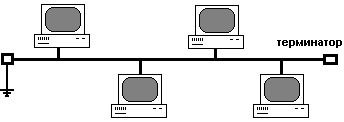
1. Простая модификация и добавление узла без нарушения целостности сети
2. Центральный коммутатор может являться средством для диагностики, мониторинга и управления сетью
3. Отказоустойчивость
4. Применение нескольких типов кабелей

Топология дерево (иерархическая звезда)

Иногда имеет смысл строить сеть с использованием нескольких концентраторов иерархически соединённых между собой связями типа звезда. Полученную структуру называют деревом. Дерево является самым распространённым типом топологии как в локальных, так и в глобальных сетях. 

Общая шина

Частным случаем звезды является конфигурация «общая шина».

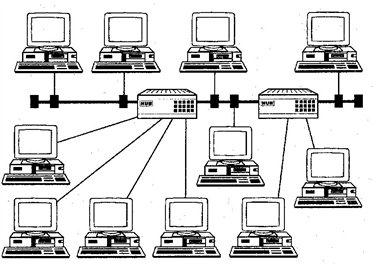


Здесь в роли центрального элемента выступает пассивный кабель, к которому по схеме монтажного ? подключается несколько компьютеров. Такую же топологию имеют многие сети, использующие беспроводную связь. В роли общей шины – общая радиосреда. Передаваемая информация доступна всем подлючённым пользователям.

Основным преимуществом такой схемы является низкая стоимость и простота наращивания. Самым серьёзным недостатком является её недостаточная надёжность. Любой дефект кабеля или какого-нибудь из многочисленных разъёмов полностью парализует всю сеть. Другой – невысокая производительность, т. к. при таком способе подключения каждый момент времени только один компьютер может передавать данные по сети. Поэтому пропускная способность канала связи всегда делится между всеми узлами сети.

Смешанная топология

В глобальных сетях мы говорим о смешанной топологии.



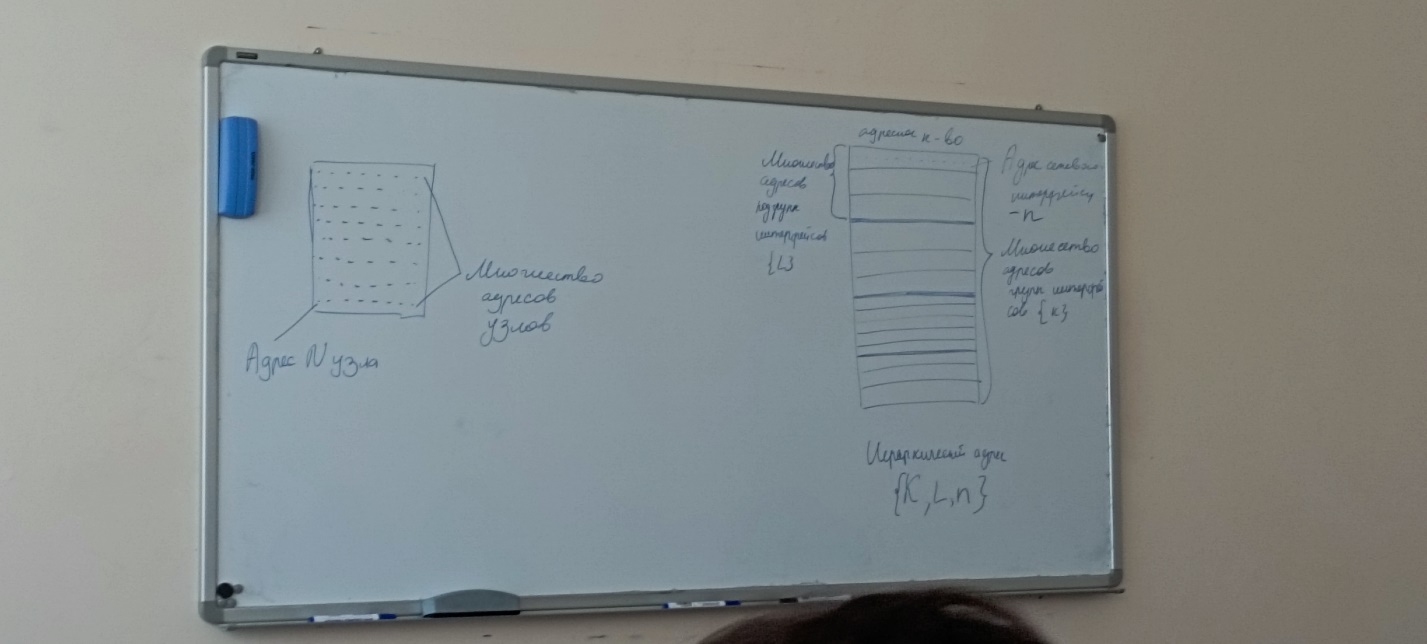
Лекция 23.09.2024

Адресация узлов в сети

Ещё одной проблемой, которую нужно учитывать при объединении трех и более компьютеров является проблема их адресации. Один компьютер может иметь несколько сетевых интерфейсов. Пример: соединение кольцо – каждый компьютер имеет минимум два интерфейса, полносвязная структура (n-1 интерфейс).

Адреса могут быть числовыми: 127.0.0.1; символьными: google.com. Могут встречаться ip-адреса в Hex: 81.1a.ff.ff. Адреса могут использоваться для идентификации не только отдельных интерфейсов, но и групп – групповые адреса. С помощью групповых адресов данные могут направляться сразу нескольким узлам.

Множество всех адресов, которые является допустимыми в рамках некой схемы адресации, называется адресным пространством. Плоское (линейное) адресное пространство и иерархическое адресное пространство:



Данная структура адресного пространства заполняется 3-мя составляющими:

1. Идентификатор группы (k)
2. Идентификатор ? (l)
3. Идентификатор узла (n) – определяет однозначно интерфейс в подгруппе

Требования к адресам сетевого интерфейса:

1. Адрес должен уникально идентифицировать свой интерфейс в сети любого масштаба
2. Схема назначения адресов должна сводить к минимуму ручной труд администратора и вероятность дублирования адресов
3. Желательно чтобы адрес имел иерархическую структуру
4. Адрес должен быть по возможности компактным, чтобы не перегружать память коммуникационной аппаратуры

Запросы на сервер:

Изображение выглядит как текст, доска, рукописный текст, стена

Автоматически созданное описание

Для преобразования адресов из одного вида в другой используются специальные вспомогательные протоколы – протокол разрешения адреса.

Пример плоской адресации – мак адрес.

Типичным представителем иерархических числовых адресов – сетевые ip и ipx адреса. В них поддерживается двухуровневая иерархия: адрес делится на старшую часть – номер сети, и младшую – номер узла. Такое разделение позволяет передавать сообщения между сетями только на основании номера сети, а номер узла используется после доставки сообщения в нужную сеть. Такой подход реализован в IPv6, предназначенный для работы в интернете.

Символьные адреса или имена предназначены для запоминания людьми и обычно несут смысловую нагрузку. Могут использоваться к в небольших, так и в крупных сетях. Для работы в больших сетях символьное имя может иметь иерархическую структуру. В современных сетях для адресации узлов применяются все три примененные схемы одновременно.

Пользователи адресую компьютеры символьными именами, которые автоматически заменяются на числовые. С помощью этих числовых номеров сообщения передаются из одной сети в другую. После доставки сообщения в сеть назначения вместо числового – символьный.

.ru -> mephi.ru -> test. -> ip -> сеть -> mac адрес

DNS – domain name system – наиболее известная система доменных имён.

IP-адреса могут использоваться для отдельных интерфейсов, группы интерфейсов, сразу всех интерфейсов в сети.

Лекция 04.10.2024

DNS-сервера

Архитектура и функционирование DNS

Чтобы организовать связь между компьютерами используются IP-адреса, но зачастую пользователь знает только имя хостинга и ничего не знает о его адресе. Следовательно, пользователю нужно получить адреса по имени хостинга. В этом ему помогает DNS, который выполняет функцию базы данных IP адресов.

DNS – система доменных имён, которая выполняет функцию передачи имени в IP-адрес, который нужен для организации связи между компьютерами в сети.

Архитектура DNS

DNS − распределенная иерархическая система серверов-имён. Большое множество баз данных, содержащих сведения о каждом домене, составляют собой DNS.

Рассмотрим на примере сайте twitch.tv, .tv − домен первого уровня, также из него можно получить адреса серверов, которые предоставляют нам дополнительную информацию о содержании поддоменов (корневая зона), Twitch – домен второго уровня, имеющий все поддомены и т. д.

Передача имени в IP-адрес происходит в несколько шагов:

Запрашиваются серверы, которые обслуживают корневую зону. Через них мы можем узнать, каким образом можно связаться с серверами, которые обслуживают домен первого уровня. От сервера первого можно узнать имена адресов второго уровня. С помощью такой архитектуры DNS можно распределить всю нагрузку между администраторами доменов.

Корневой уровень DNS

Корневые серверы DNS являются главной составляющей системы, т. к. поддерживают доступ к корневой зоне DNS. Корневая зона имеет сведения обо всех доменах наивысшего уровня. Например, национальные домены «.ru», «.en»; общего назначения – «.com» и спонсируемые «.museum».

Эта инфа используется для того, чтобы узнать на какие серверы DNS следует послать следующий запрос. Т. е. любой новый (несохраненный в кэше клиента) запрос имеет вначале запрос имеет обращение к новому серверу.

Самый первый запрос, который реализуется клиентом, производится по IP адресу сервера, а не по его имени. Т. к. изначальную информацию о корневых серверах и их адреса невозможно получить из DNS, то для хранения этой информации предназначен специальный файл – «hints», который находится у клиента и обычно его можно получить вместе с ПО.

Функционирование DNS

DNS сервер обеспечивает разрешение имён в сетях, основанных на протоколах TCP/IP. Это даёт возможность пользователям использовать для идентификации удаленных узлов имена, а не IP-адреса. Компьютер клиента направляет имя DNS-серверу, а тот сообщает IP.

Если в БД нет информации, которая соответствует удалённому узлу, то DNS-сервер возвратит клиенту адрес другого DNS-сервера, который, вероятно, даст ему нужный IP. До тех пор, пока клиент не получит нужный IP адрес или если не окажется, что указанное имя не относится к узлу в конкретном пространстве имен DNS процесс продолжается рекурсивно.

С помощью DNS можно приобрести следующие сведения:

1. IP-адрес хоста
2. Доменное наименование хоста
3. Данные о псевдониме хоста, тип ЦП и ОС хоста
4. Сетевые протоколы, которые поддерживает хост
5. Почтовый шлюз
6. Почтовый ящик
7. Почтовую группу
8. IP-адрес и доменное имя сервера имён доменов

DNS ACTIVE DIRECTORY

Служба DNS-сервер встроена в структуру и реализацию службы каталогов Active Directory (AD). Служба каталогов AD представляет собой инструмент организации, управления и выбора местоположения в сети на уровне предприятия.

После установки AD на сервер роль этого сервера возрастает до уровня контролёра указанного домена.

21.10.2024

Модель OSI

Тезис о пользе стандартизации в сетях приобретает особое значение: суть сети это соединение разного оборудования, а значит проблема совместимости является одной из наиболее острых. В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия. Именно на основе этого подхода была создана стандартная семиуровневая модель взаимодействия открытых систем (OSI).

Многоуровневый подход. Декомпозиция задач сетевого взаимодействия

Для решения сложных задач, которой является взаимодействие устройств в сети, используется специальный приём – декомпозиция, т. е. разбиение одной задачи на несколько задач-модулей. Декомпозиция состоит в четком определении функций каждого модуля и порядка их взаимодействий (интерфейсов). В результате достигается логическое упрощение задач и появляется возможность модификации отдельных модулей без изменения остальной части системы.

При декомпозиции часто используют многоуровневый подход:

1. Всё множество модулей, решающих частные задачи, разбивают на группы и упорядочивают по уровню, образующим иерархию.
2. В соответствии с принципами иерархии для каждого промежуточного уровня можно указать непосредственно примыкающие к нему вышележащие и нижележащие к нему.
3. Группа модулей, составляющих каждый уровень, должна быть сформирована таким образом, чтобы все модули этой группы для выполнения своих задач обращались с запросами только к модулю соседнего нижележащего уровня.
4. Результаты работы всех модулей, отнесённых к некоторому уровню, могут быть переданы только модулем соседнего вышележащего уровня.

Изображение выглядит как текст, доска, рукописный текст, стена

Автоматически созданное описание

Такая иерархическая декомпозиция задачи предполагает четкое определение функций каждого уровня и интерфейса между уровнями. Интерфейс определяет набор функций, которые нижележащий уровень представляет вышележащему. В результате иерархической декомпозиции достигается относительная независимость уровней, а значит возможность их автономной разработки и модификации.

Средства решения задачи организации сетевого взаимодействия могут быть представлены в виде иерархически организованного множества модулей. Пример: модулям нижнего уровня можно поручить вопросы связанные с надёжной передачей информации между двумя соседними узлами, а модулям следующего, более высокого уровня – транспортировку сообщений в пределах всей сети.

Протокол, интерфейс и stack-протокол

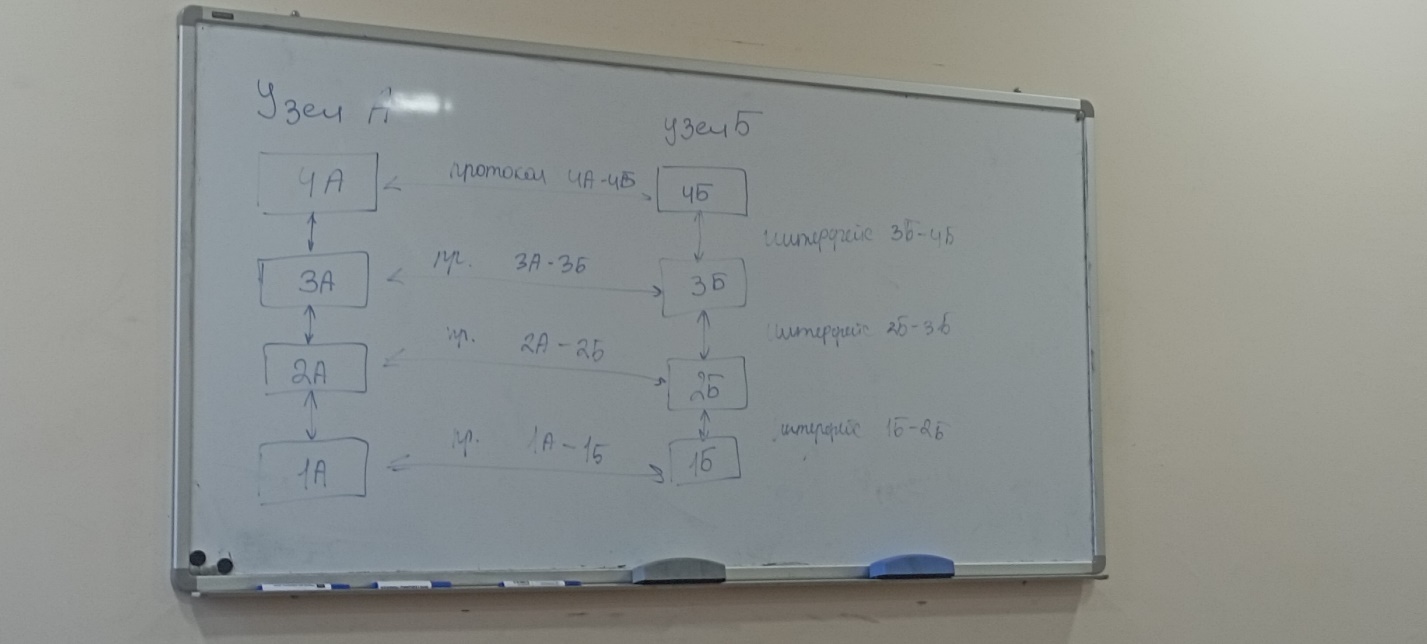
Многоуровневое представление средств сетевого взаимодействия имеет свою специфику, связанную с тем, что в процессе обмена сообщениями участвуют две стороны. Т. е. в данном случае необходимо организовать согласованную работу двух иерархий. Оба участника сетевого обмена должны принять множество соглашений. Например, уровень и форму электрических сигналов, способ определения длины сообщений; договориться о методах контроля и достоверности.

Рис − Взаимодействие двух узлов

С каждой стороны средства взаимодействия представлены 4-мя уровнями. Процедура взаимодействия этих двух узлов может быть описана в виде набора правил взаимодействия каждой пары соответствующего уровня. Формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются протоколом.

Модули, реализующие протоколы соседних уровней и находящиеся в одном узле, также взаимодействуют друг с другом в соответствии с помощью определенных правил. Эти правила принято называть интерфейсом. Интерфейс определяет последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на соседних уровнях в одном узле. Интерфейс определяет набор услуг, предоставляемых данным уровнем соседнему уровню. Иерархически организованный набор протоколов достаточный … − называется stack-протоколом.

Коммуникационные протоколы могут быть реализованы как программно, так и аппаратно. Протоколы нижних уровней часто реализуются комбинацией программных и аппаратных средств, а протоколы верхних уровней, как правило, чисто программными средствами.

Протоколы реализуются не только компьютерами, но и другими сетевыми устройствами – концентраторами, мостами, коммутаторами, маршрутизаторами и т. д.