МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВПО

**«ПСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

К а ф е д р а

“ И н ф о р м а ц и о н н ы е с и с т е м ы и т е х н о л о г и и ”

Контрольная работа

по учебной дисциплине

«ТЕОРИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И СИСТЕМ»

(ВАРИАНТ № 12)

Выполнил:

студент группы 1022-03

шифр 161 12092

Ковалевский Р. А.

Проверил преподаватель:

Л.В.Мотайленко

Псков

2018

# Оглавление

[Извлечение информации 3](#_Toc516144836)

[Объектно-ориентированный подход 4](#_Toc516144837)

[Объектно-ориентированная декомпозиция 5](#_Toc516144838)

[Транспортирование информации 7](#_Toc516144839)

[Требования к компьютерным сетям 11](#_Toc516144840)

[Производительность 12](#_Toc516144841)

[Надежность и безопасность 15](#_Toc516144842)

[Список используемой литературы: 18](#_Toc516144843)

[Приложение 1. 19](#_Toc516144844)

# Извлечение информации

Источниками данных в любой предметной области являются объекты и их свойства, процессы и функции, выполняемые этими объектами или для них. Любая предметная область рассматривается в виде трех представлений.

По аналогии с добычей полезных ископаемых процесс извлечения информации направлен на получение ее наибольшей концентрации. В связи с этим процесс извлечения можно представить, как прохождение информации через трехслойный фильтр, в котором осуществляется оценка синтаксической ценности (правильность представления), семантической (смысловой) ценности, прагматической (потребительской) ценности.

При извлечении информации важное место занимают различные формы и методы исследования данных:

1.поиск ассоциаций, связанных с привязкой к какому-либо событию;

2. обнаружение последовательностей событий во времени;

3.выявление скрытых закономерностей по наборам данных, путем определения причинно-следственных связей между значениями определенных косвенных параметров исследуемого объекта (ситуации, процесса);

4.оценка важности (влияния) параметров на развитие ситуации;

5.классифицирование (распознавание), осуществляемое путем поиска критериев, по которым можно было бы относить объект (события, ситуации, процессы) к какой-либо категории;

6.кластеризация, основанная на группировании объектов по каким-либо признакам;

7.прогнозирование событий и ситуаций.

Следует упомянуть неоднородность информационных ресурсов, характерную для многих предметных областей. Одним из путей решения данной проблемы является объектно-ориентированный подход, наиболее распространенный в настоящее время.

# Объектно-ориентированный подход

Основные идеи объектно-ориентированного подхода опираются на следующие положения:

1.программа представляет собой модель некоторого реального процесса, части реального мира;

2.модель реального мира или его части может быть описана как совокупность взаимодействующих между собой объектов;

3.объект описывается набором параметров, значения которых определяют состояние объекта, и набором операций (действий), которые может выполнять объект;

4.взаимодействие между объектами осуществляется посылкой специальных сообщений от одного объекта к другому. Сообщение, полученное объектом, может потребовать выполнения определенных действий, например, изменения состояния объекта;

5.объекты, описанные одним и тем же набором параметров и способные выполнять один и тот же набор действий, представляют собой класс однотипных объектов.

С точки зрения языка программирования класс объектов можно рассматривать как тип данного, а отдельный объект - как данное этого типа. Определение программистом собственных классов объектов для конкретного набора задач должно позволить описывать отдельные задачи в терминах самого класса задач (при соответствующем выборе имен типов и имен объектов, их параметров и выполняемых действий).

Таким образом, объектно-ориентированный подход предполагает, что при разработке программы должны быть определены классы используемых в программе объектов и построены их описания, затем созданы экземпляры необходимых объектов и определено взаимодействие между ними.

# Объектно-ориентированная декомпозиция

Объектно-ориентированная декомпозиция - это разбиение системы на сущности, являющиеся какими-либо объектами действующими в той ситуации, которую как раз и моделирует система.

Такая декомпозиция считается "более продвинутой" (при разработке сложных систем) в отличии от Алгоритмической -

при которой просто происходит разбиение функционала - ни по отношению к сущности (типа - набор методов в данном классе реализует функционал описываемого объекта), а просто "сверху вниз" - то есть об общего к частному.

Например при алгоритмической композиции - в случае если требуется описать процесс кормления коровы - никто не будет создавать два класса, например:

-корова + человек

- как в случае объектно-ориентированной декомпозиции, а просто начнут уточнять функционал, например так (вначале написав базовую функцию):

1) покормить корову()

2) купить корм () + покормить корову()

3) купить корм () + насыпать корм в кормушку() + убедится что корова начала есть()

то есть при алгоритмической декомпозиции функционал просто уточняется и разбивается на части - без относительно того к каким объектам он имеет отношения или какими объектами данные функции выполняются.

Декомпозиция на базе объектно-ориентированного подхода основана на выделении следующих базовых понятий: объект, класс, экземпляр.

Объект - это абстракция множества предметов реального мира, обладающих одинаковыми характеристиками и законами поведения.Объект характеризует собой типичный неопределенный элемент такого множества. Основной характеристикой объекта является состав его атрибутов (свойств).

Атрибуты - это специальные объекты, посредством которых можно задать правила описания свойств других объектов.

Экземпляр объекта - это конкретный элемент множества. К примеру, объектом может являться государственный номер автомобиля, а экземпляром этого объекта - конкретный номер К173 ПА.

Класс-это множество предметов реального мира, связанных общностью структуры и поведением. Элемент класса - это конкретный элемент данного множества. К примеру, класс регистрационных номеров автомобиля.

Обобщая эти определения, можно сказать, что объект - это типичный представитель класса, а термины “экземпляр объекта” и “элемент класса” равнозначны.

# Транспортирование информации

Основным физическим способом реализации операции транспортировки является использование локальных сетей и сетей передачи данных. При разработке и использовании сетей для обеспечения совместимости используется ряд стандартов, объединенных в семиуровневую модель открытых систем, принятую во всем мире и определяющую правила взаимодействия компонентов сети на данном уровне (протокол уровня) и правила взаимодействия компонентов различных уровней (межуровневый интерфейс). Meждународные стандарты в области сетевого информационного обмена нашли отражение в эталонной семиуровневой модели, известной как модель OSI (Open System Intercongtction - связь открытых систем) (Приложение 1). Данная модель разработана международной организацией по стандартизации (International Standards Organization - ISO). Большинство производителей сетевых программно-аппаратных средств стремятся придерживаться модели OSI. Но в целом добиться полной совместимости пока не удается.

*Физический уровень* реализует физическое управление и отно­сится к физической цепи,

*Канальный уровень.* На этом уровне осуществляется управление звеном сети (каналом) и реализуется пересылка блоков (совокуп­ности битов) информации по физическому звену.

*Сетевой уровень* относится к виртуальной (воображаемой) цепи, которая не обязана существовать физически. С помощью интер­фейса, обеспечиваемого этим уровнем, удается «спрятать» сложно­сти управления передачей на физическом уровне.

*Транспортный уровень.* На транспортном уровне контролируется очередность пакетов со­общений и их принадлежность.

*Сеансовый уровень.* На данном уровне координируются и стандартизируются процессы ус­тановления сеанса, управления передачей и приемом пакетов сооб­щений, завершения сеанса.

*Управление представлением.* Программные средства этого уровня выполняют преобразования данных из внутреннего формата пере­дающего компьютера во внутренний формат компьютера-получате­ля, если эти форматы отличаются друг от друга (например, IBM PC и DEC).

*Прикладной уровень* относится к функциям, которые обеспечи­вают поддержку пользователю на более высоком прикладном и системном уровнях, например:

• организация доступа к общим сетевым ресурсам: информа­ции, дисковой памяти, программным приложениям, внешним уст­ройствам (принтерам, стримерам и др.);

• общее управление сетью (управление конфигурацией, разгра­ничение доступа к общим ресурсам сети, восстановление работо­способности после сбоев и отказов, управление производительно­стью);

• передача электронных сообщений, включая электронную почту;

*•* организация электронных конференций;

• диалоговые функции высокого уровня.

Модель OSI представляет собой стандартизированный каркас и общие рекомендации, требования же к конкретным компонентам сетевого программного обеспечения задаются протоколами.

Протоколы сетевого взаимодействия можно классифицировать по степени близости к физической среде передачи данных. Это протоколы:

• нижнего уровня, распространяемые на канальный и физиче­ский уровни модели OSI;

• среднего уровня, распространяемые на сетевой, транспорт­ный и сеансовый уровни OSI;

• верхнего уровня, распространяемые на уровень представле­ния и прикладной уровень модели OSI.

При каждой реализации протоколов вышестоящих уровней ис­пользуются реализации протоколов нижестоящих уровней.

Протоколы нижнего уровня OSI соответствуют уровню сетевых аппаратных средств и нижнему уровню сетевого программного обеспечения. Среди наиболее распространенных стандартов данно­го уровня выделим следующие :

• стандарт NDIS (Network Driver Interface Specification - спе­цификация интерфейса сетевых драйверов), разработанный совме­стно фирмами Microsoft и 3Com;

• стандарт ODI (Open Datalink Interface - открытый интерфейс связи), разработанный совместно фирмами Novell и Apple Computer.

Протоколы среднего уровня распространяются на сетевой, транспортный и сеансовый уровни эталонной модели. По типу межкомпьютерного обмена эти протоколы можно классифициро­вать следующим образом:

• сеансовые протоколы (протоколы виртуального соединения);

• дейтаграммные протоколы.

Наиболее часто используемыми наборами протоколов среднего уровня являются следующие:

• набор протоколов SPX/IPX, используемый в локальных се­тях, функционирующих под управлением сетевой операционной системы NetWare;

• протоколы NetBIOS и NetBEUI, поддерживаемые большин­ством сетевых операционных систем и используемые только в ло­кальных сетях;

• протоколы TCP/IP, являющиеся стандартом для глобальной сети Internet, используемые в локальных сетях и поддерживаемые большинством сетевых операционных систем.

Протоколы верхнего уровня соответствуют уровню пользовате­лей и прикладных программ и распространяются на уровень пред­ставления и прикладной уровень эталонной модели сетевого взаи­модействия. Наиболее распространенными являются следующие высокоуровневые протоколы:

• перенаправления запросов и обмена сообщениями (SMB, NCP);

• управления сетями (SNMP);

• сетевой файловой системы (NFS);

• вызова удаленных процедур (RPC);

• повышающие эффективность использования протоколов TCP/IP среднего уровня (DNS, DHSP);

• удаленного доступа к компьютерным ресурсам (SLIP, PPP, Telnet);

• передачи файлов (FTP);

• передачи гипертекста (HTTP);

• электронной почты (SMTP, POP3, IMAP4);

• организации электронных конференций и системы новостей (NNTP).

# Требования к компьютерным сетям

Обсуждаются важнейшие показатели работы сети: производительность, надежность и безопасность, расширяемость и масштабируемость, прозрачность, поддержка разных видов трафика, характеристики качества обслуживания, управляемость и совместимость.

Соответствие стандартам - это только одно из многих требований, предъявляемых к современным сетям. В этом разделе мы остановимся на некоторых других, не менее важных.

Самое общее пожелание, которое можно высказать в отношении работы сети - это выполнение сетью того набора услуг, для оказания которых она предназначена: например, предоставление доступа к файловым архивам или страницам публичных Web-сайтов Internet, обмен электронной почтой в пределах предприятия или в глобальных масштабах, интерактивный обмен голосовыми сообщениями IP-телефонии и т.п.

Все остальные требования - производительность, надежность, совместимость, управляемость, защищенность, расширяемость и масштабируемость - связаны с качеством выполнения этой основной задачи. И хотя все перечисленные выше требования весьма важны, часто понятие "качество обслуживания" (Quality of Service, QoS) компьютерной сети трактуется более узко: в него включаются только две самые важные характеристики сети - производительность и надежность.

## *Производительность*

Потенциально высокая производительность - это одно из основных преимуществ распределенных систем, к которым относятся компьютерные сети. Это свойство обеспечивается принципиальной, но, к сожалению, не всегда практически реализуемой возможностью распределения работ между несколькими компьютерами сети.

Основные характеристики производительности сети:

1.время реакции;

2.скорость передачи трафика;

3.пропускная способность;

4.задержка передачи и вариация задержки передачи.

Время реакции сети является интегральной характеристикой производительности сети с точки зрения пользователя. Именно эту характеристику имеет в виду пользователь, когда говорит: "Сегодня сеть работает медленно".

В общем случае время реакции определяется как интервал между возникновением запроса пользователя к какой-либо сетевой службе и получением ответа на него.

Очевидно, что значение этого показателя зависит от типа службы, к которой обращается пользователь, от того, какой пользователь и к какому серверу обращается, а также от текущего состояния элементов сети - загруженности сегментов, коммутаторов и маршрутизаторов, через которые проходит запрос, загруженности сервера и т.п.

Поэтому имеет смысл использовать также и средневзвешенную оценку времени реакции сети, усредняя этот показатель по пользователям, серверам и времени дня (от которого в значительной степени зависит загрузка сети).

Время реакции сети обычно складывается из нескольких составляющих. В общем случае в него входит:

время подготовки запросов на клиентском компьютере;

время передачи запросов между клиентом и сервером через сегменты сети и промежуточное коммуникационное оборудование;

время обработки запросов на сервере;

время передачи ответов от сервера клиенту и время обработки получаемых от сервера ответов на клиентском компьютере.

Очевидно, что разложение времени реакции на составляющие пользователя не интересует - ему важен конечный результат. Однако для сетевого специалиста очень важно выделить из общего времени реакции составляющие, соответствующие этапам собственно сетевой обработки данных, - передачу данных от клиента к серверу через сегменты сети и коммуникационное оборудование.

Знание сетевых составляющих времени реакции позволяет оценить производительность отдельных элементов сети, выявить узкие места и при необходимости выполнить модернизацию сети для повышения ее общей производительности.

Производительность сети может характеризоваться также скоростью передачи трафика.

Скорость передачи трафика может быть мгновенной, максимальной и средней.

средняя скорость вычисляется путем деления общего объема переданных данных на время их передачи, причем выбирается достаточно длительный промежуток времени - час, день или неделя;

мгновенная скорость отличается от средней тем, что для усреднения выбирается очень маленький промежуток времени - например, 10 мс или 1 с;

максимальная скорость - это наибольшая скорость, зафиксированная в течение периода наблюдения.

Пропускная способность - максимально возможная скорость обработки трафика, определенная стандартом технологии, на которой построена сеть. Пропускная способность отражает максимально возможный объем данных, передаваемый сетью или ее частью в единицу времени.

Пропускная способность уже не является, подобно времени реакции или скорости прохождения данных по сети, пользовательской характеристикой, так как она говорит о скорости выполнения внутренних операций сети - передачи пакетов данных между узлами сети через различные коммуникационные устройства. Зато она непосредственно характеризует качество выполнения основной функции сети - транспортировки сообщений - и поэтому чаще используется при анализе производительности сети, чем время реакции или скорость.

Пропускная способность измеряется либо в битах в секунду, либо в пакетах в секунду.

Пропускная способность сети зависит как от характеристик физической среды передачи (медный кабель, оптическое волокно, витая пара) так и от принятого способа передачи данных (технология Ethernet, FastEthernet, ATM). Пропускная способность часто используется в качестве характеристики не столько сети, сколько собственно технологии, на которой построена сеть. Важность этой характеристики для сетевой технологии показывает, в частности, и то, что ее значение иногда становится частью названия, например, 10 Мбит/с Ethernet, 100 Мбит/с Ethernet.

В отличие от времени реакции или скорости передачи трафика пропускная способность не зависит от загруженности сети и имеет постоянное значение, определяемое используемыми в сети технологиями.

Пропускная способность и задержки передачи также являются независимыми параметрами, так что сеть может обладать, например, высокой пропускной способностью, но вносить значительные задержки при передаче каждого пакета. Пример такой ситуации дает канал связи, образованный геостационарным спутником. Пропускная способность этого канала может быть весьма высокой, например 2 Мбит/с, в то время как задержка передачи всегда составляет не менее 0,24 с, что определяется скоростью распространения электрического сигнала (около 300000 км/с) и длиной канала (72000 км).

## *Надежность и безопасность*

Одна из первоначальных целей создания распределенных систем, к которым относятся и вычислительные сети, состояла в достижении большей надежности по сравнению с отдельными вычислительными машинами.

Важно различать несколько аспектов надежности.

Для сравнительно простых технических устройств используются такие показатели надежности, как:

среднее время наработки на отказ;

вероятность отказа;

интенсивность отказов.

Однако эти показатели пригодны для оценки надежности простых элементов и устройств, которые могут находиться только в двух состояниях - работоспособном или неработоспособном. Сложные системы, состоящие из многих элементов, кроме состояний работоспособности и неработоспособности, могут иметь и другие промежуточные состояния, которые эти характеристики не учитывают.

Для оценки надежности сложных систем применяется другой набор характеристик:

готовность или коэффициент готовности;

сохранность данных;

согласованность (непротиворечивость) данных;

вероятность доставки данных;

безопасность;

отказоустойчивость.

Готовность или коэффициент готовности (availability) означает период времени, в течение которого система может использоваться. Готовность может быть повышена путем введения избыточности в структуру системы: ключевые элементы системы должны существовать в нескольких экземплярах, чтобы при отказе одного из них функционирование системы обеспечивали другие.

Чтобы компьютерную систему можно было считать высоконадежной, она должна как минимум обладать высокой готовностью, но этого недостаточно. Необходимо обеспечить сохранность данных и защиту их от искажений. Кроме того, должна поддерживаться согласованность (непротиворечивость) данных, например если для повышения надежности на нескольких файловых серверах хранится несколько копий данных, то нужно постоянно обеспечивать их идентичность.

Другим аспектом общей надежности является безопасность (security), то есть способность системы защитить данные от несанкционированного доступа. В распределенной системе это сделать гораздо сложнее, чем в централизованной. В сетях сообщения передаются по линиям связи, часто проходящим через общедоступные помещения, в которых могут быть установлены средства прослушивания линий. Другим уязвимым местом могут стать оставленные без присмотра персональные компьютеры. Кроме того, всегда имеется потенциальная угроза взлома защиты сети от неавторизованных пользователей, если сеть имеет выходы в глобальные общедоступные сети.

Еще одной характеристикой надежности является отказоустойчивость (fault tolerance). В сетях под отказоустойчивостью понимается способность системы скрыть от пользователя отказ отдельных ее элементов. Например, если копии таблицы базы данных хранятся одновременно на нескольких файловых серверах, пользователи могут просто не заметить отказа одного из них. В отказоустойчивой системе выход из строя одного из ее элементов приводит к некоторому снижению качества ее работы (деградации), а не к полному останову. Так, при отказе одного из файловых серверов в предыдущем примере увеличивается только время доступа к базе данных из-за уменьшения степени распараллеливания запросов, но в целом система будет продолжать выполнять свои функции.

# Список используемой литературы:

# Иванова Г.С., Ничушкина Т.Н., Пугачев Е.К. Объектноориентированное программирование: Учеб. для вузов/ Под ред. Г.С. Ивановой. - М.: Изд-во МГТУ им. Н.Э. Баумана, 2001. - 320 с., ил. (Сер. Информатика в техническом университете). ISBN 5-7038-1525-8

# Ввод и обработка цифровой информации. Практикум. Учебное пособие: А. В. Курилова, В. О. Оганесян — Москва, Академия, 2013 г.- 160 с.

# Информатика. Базовый курс // Под ред. С.В. Симоновича, СПб., 2000.

# Бадд Т. Объектно-ориентированное программирование в действии: Пер. с англ. СПб.: Питер, 1997. 464 с.

# Иванова, Г. С. Объектно-ориентированное программирование / Г.С. Иванова, Т.Н. Ничушкина, Е.К. Пугачев. - М.: МГТУ им. Н. Э. Баумана, 2003. - 368 c

# Кузин, А.В. Компьютерные сети: Учебное пособие / А.В. Кузин.. - М.: Форум, НИЦ ИНФРА-М, 2013. - 192 c.

# Руководство по технологиям объединённых сетей. 4-е изд. — М.: Вильямс, 2005. ISBN 5-8459-0787-X.

# А. Филимонов. Построение мультисервисных сетей Ethernet. — М.: BHV, 2007. ISBN 978-5-9775-0007-4.

# Луганцев, Л.Д. Компьютерные сети / Л.Д. Луганцев. - М.: МГУИЭ, 2001. - 452 c.

# Приложение 1.

****