**Лекция № 11 (28)**– 10.11.23 г. Программы. Интеллектуальная доминанта

### **8.1.6.1.8. Программы**

Примем в общем виде:

**Программа** (греч. programma – объявление, распоряжение) – однозначно установленная кому-то на понятном ему языке обязательная последовательность каких-либо логических и/или физических действий для достижения нужных целей.

Программа может предназначаться человеку или какому-либо техническому устройству, например, стиральной машине, микроволновой печи и т. п. и, наконец, компьютеру или вычислительной сети. В последнем случае программа называется компьютерной или сетевой программой. Наряду с этим используются термины “программное средство”, “программный продукт”, “программное изделие”. Их следует считать синонимами, но, в отличие от программы (программного средства), программное изделие (программный продукт) снабжается комплектом эксплуатационной и правовой документации, которая позволяет пользователю самостоятельно и свободно эффективно применять изделие по назначению.

В функционирующей системе используется не одна, а несколько взаимодополняющих друг друга компьютерных программ (далее по тексту просто программ, являющихся по умолчанию компьютерными или сетевыми), которые в совокупности образуют программное обеспечение системы. Более точно:

**Программное обеспечение** – упорядоченное множество программ, обеспечивающее функционирование автоматизированной информационной системы.

К настоящему времени известно довольно много конкретных программ, и их количество постоянно увеличивается. Однако безотносительно конкретизации программ программное обеспечение состоит из общего программного обеспечения (**ОПО**) и специального программного обеспечения (**СПО**). Специальное программное обеспечение включает программные средства решения задач пользователе **ПСЗДЧ**), т.е. реализующие прикладные технологии и задачи пользователей, рассмотренные выше, и программные средства защиты системы (**ПСЗЩТ**), т.е. реализующие системную технологию, соответствующую этому. В состав ПСЗАЩ входят программные средства шифрования (**ПСШФР**), авторизации (**ПСАВТ**), аутентификации (**ПСАУТ**), цифровой электронной подписи (**ПСЦЭП**). Специальное программное обеспечение, как правило, уникально и каждый раз для иного применения разрабатывается заново полностью или частично, возможно, при использовании универсального средства (продукта) уникально настраивается.

Иное дело общее программное обеспечение, которое инвариантно, т.е. не чувствительно, к специфике новой системы, и потому квазипостоянен его базовый состав, предопределённый очевидным логическим контекстом. Действительно, в общем случае необходимо:

управлять телекоммуникациями, т.е. вводом информации в каналы связи, передачей её по каналам связи и выводом информации из каналов связи,

управлять базой данных,

управлять вычислительным процессом с учётом вовлечённости в него техники, программ, информации, людей (пользователей),

формировать вторичную информацию,

выявлять латентную информацию,

отображать информацию на средствах отображения коллективного пользования,

информационно поддерживать пользователей при принятии ими управленческих решений,

предоставлять пользователям возможность пользоваться в качестве рабочих инструментов комплексными функциями (электронная почта, электронный документооборот, электронная цифровая подпись, электронная доска объявлений, аудиовидеоконференцсвязь и т.п.),

оснащать рабочие места пользователей средствами для выполнения рутинной работы в автоматизированном режиме,

обеспечивать пользователям выход в Интернет и доступ к облачным вычислениям, услугам блокчейна, биткоина и т.п.,

наконец, синхронизировать во времени работу многочисленных вычислительных средств, упорядоченных принятой топологией из числа типовых топологий, рассмотренных ранее, и реализуемых на их множестве разнообразных процессов.

Отсюда следует базовый состав программного обеспечения информационно-технологической доминанты:

операционные системы сетевая (**СОС**) и локальные (**ОС**) в узлах принятой топологии,

программные средства управления телеобработкой (**СУТ**),

программные средства управления базой данных (**СУБД** – система управления базой данных, что является общепринятым обозначением),

программные средства управления вычислительным процессом (**СУВП** – система управления вычислительным процессом),

программные средства хранилища данных (**ПСХД**),

программные средства центра данных (**ПСЦД**),

программные средства поддержки принятия решений (**ПСППР**),

программные средства отображения информации коллективного пользования (**ПСОИКП**),

программные средства реализации комплексных функций (**ПСРКФ**),

программные средства пользователей (**ПСП**) на автоматизированных рабочих местах,

программные средства облачных вычислений (**ПСОВ**),

программные средства доступа в интернет (**ПСДИН**),

программные средства блокчейна (**ПСБЛ**),

программные средства биткоина (**ПСБТ**),

программные средства защиты системы, перечисленные выше.

Базовый состав программного обеспечения в силу привходящих обстоятельств может быть расширен или, напротив, урезан. Но в любом случае программное средство должно быть представлено на языке, доступном компьютеру. Поэтому рассмотрим языки программирования.

### **8.1.6.1.9. Языки программирования**

В общем виде:

**Язык** – знаковая система, являющаяся носителем смыслов.

Здесь:

**Смысл** – содержание как единство внутренних свойств и отношений чего-либо, позволяющее отличить его от всего другого.

Язык позволяет создавать разнообразные языковые конструкции (лат. construction– составление, построение), подчинённые лексическим, синтаксическим и семантическим правилам. Разнообразие языковых конструкций представляется множеством слов, словосочетаний, предложений и текстов различной длины.

Здесь:

**Лексика** (греч. lexikos – относящийся к слову) – словарь языка, т.е. пригодные или общепринятые для употребления слова языка.

**Синтаксис** (греч. syntaxis– построение, порядок) – правила построения конструкций языка.

**Семантика** (греч.semanticos– обозначающий) – смысл конструкций языка.

Любой язык, будучи знаковой системой, имеет алфавит (от названия первых двух греческих букв: альфа и бета, трансформировавшейся в новогреч. – вита). Здесь:

**Алфавит** - разрешенный к применению набор символов, из которых могут быть образованы различные языковые конструкции.

Языки бывают разными: естественными, искусственными или формальными, в том числе и языками программирования.

Естественный язык является средством человеческого общения, связанного с мышлением. Потребовалось немало времени, чтобы человеческий разум научился обозначать окружающие предметы, связи меду ними, действия с ними и передавать это друг другу с помощью языковых средств. Развитие речи поначалу шло очень медленно, а письменные формы и возможность передачи отвлечённых понятий появились в истории человечества очень поздно, возможно не ранее, чем четыреста или пятьсот поколений назад [1].

Современные естественные языки насчитывают многие тысячи слов: словарь Владимира Даля содержит 200 тысяч слов, по данным Интернета в современном английском языке – примерно 750 тысяч слов, внемецком, по разным подсчётам, - от 185 до 300 тысяч слов. Словарный запас А.С. Пушкина, как и У. Шекспира, составлял примерно 20 тысяч слов. Имеются данные о количестве употребляемых слов и у других писателях: Гоголь (только в “Мёртвых душах”) – 10 000 слов, Есенин – около 19 000. Словарный состав современного образованного человека оценивается в среднем в 3 – 5 тысяч слов.

Французский современник Пушкина Проспер Мериме писал [ ]:“Русский язык, насколько я могу судить о нём, является богатейшим из всех европейских наречий и кажется нарочно созданным для выражения тончайших оттенков. Одарённый чудесной сжатостью, соединённый с ясностью, он довольствуется одним словом для передачи мысли, когда другому языку потребовались бы для этого целые фразы”. Поистине, это про “великий и могучий”. Но состояние русского языка по итогам XX века вызывает тревогу [2]: обогащение русского языка последних десятилетий происходит за счёт усвоения и переделки иностранных слов и понятий, свои ‘’родные” неологизмы – это в абсолютном большинстве жаргонизмы и вульгаризмы и внутреннее ощущение богатства русского языка оказывается не соответствующим его фактической бедности. Удручающе засилье русского языка транслитерациями англоязычного компьютерного жаргона.

Здесь:

**Неологизмы** (греч. neos – новый + греч. logos– слово) – новые слова и выражения для обозначения чего-то, необычность которых ясно ощущается носителями языка.

Естественные языки являются продуктом тысячелетних мировых цивилизационных процессов. Иное дело – искусственные языки, создаваемые человеком в угоду своим текущим потребностям, и к которым относятся языки программирования.

**Язык программирования** – это формальный язык написания программ, исполняемых компьютером, точнее, определяющих заданное функционирование компьютера или множества компьютеров.

К настоящему времени известно порядка 8 тысяч языков программирования, их количество растает и каждый из них имеет своё предназначение и область применения, свои достоинства и недостатки [3].

Компьютерная программа представляет собой упорядоченную последовательность команд, предназначенных для управления работой компьютера. Процессор компьютера (большая интегральная схема) получает информацию, которую следует преобразовывать (обрабатывать), и управляющую информацию, которая указывает какие именно преобразования над обрабатываемой информацией нужно осуществлять, в виде электрических сигналов. Наличие электрического сигнала на входе процессора означает ‘’1”, его отсутствие – “0”. Единица или ноль является битом. В итоге входная информация (обрабатываемая и управляющая) представляется последовательностью комбинаций из нолей и единиц (комбинация из восьми битов образует байт). Любая такая комбинация называется двоичным или машинным кодом, а множество кодов и правила образования из них нужных последовательностей– машинным языком.

Писать программы в машинных кодах или на машинном языке очень сложно, причём с ростом размера программ эта задача кратно усложняется. Поэтому, относительно вскоре после появления первых промышленных моделей электронных вычислительных машин и начала их активного применения, усилиями талантливых учёных и программистов разработка компьютерных программ для решения нужных задач приобрела характер, соответствующий следующей диаграмме []:

**Естественный язык**→

**Математический язык** →

**Язык схем алгоритмов** →

**Алгоритмический язык** →

**Машинный язык**, фиксирующей пять этапов создания программы.

Сначала подлежащая решению задача формулируется на естественном языке и для неё находится аналитически математическое решение в виде формулы или системы уравнений, которые преобразуются к виду блок-схемы алгоритма. Далее получается описание блок-схемы алгоритма на алгоритмическом языке, которое, наконец, переводится на машинный язык, понятный компьютеру. Достоинство диаграммы заключено в том, что первый этап реализуется на естественном языке и последующие три – на его примитивах, интуитивно и реально без особого интеллектуального напряжения понятных человеку, а преобразование алгоритмического вида программы в машинный код осуществляется автоматически, т.е. без участия человека, с помощью заранее разработанных для этого программ-трансляторов.

Здесь:

**Примитив** (лат. primitivus – первый, самый ранний) – нечто упрощённое.

Существует два вида программ-трансляторов: компиляторы и интерпретаторы. Компилятор переводит программу в машинный код сразу и целиком, а интерпретатор – последовательными частями.

Таким образом преодоление сложности разработки программ и программных систем свелось к созданию алгоритмических языков (в большей степени) и трансляторов (в меньшей степени). Неравнозначность в данном случае объясняется тем, что множество и разнообразие подлежащих решению на компьютере задач и пользователей алгоритмических языков существенно перекрывает множество и разнообразие компьютеров, особенности которых должны учитывать трансляторы.

Таким образом, по сути, язык программирования – это алгоритмический язык плюс транслятор, но зачастую контекстно язык программирования и алгоритмический язык являются синонимами. Тем не менее:

**Алгоритмический язык** – набор символов (алфавит) и правила составления (синтаксис) и истолкования (семантика) языковых конструкций из этих символов для записи алгоритмов.

**Алгоритм** или **Алгорифм** (от algorithmi, algorismus– лат. транслитерация имени математика аль-Хорезми) – последовательность действий для достижения цели или решения задачи, точнее, вычислительной задачи.

Последнее уточнение важно в том смысле [ ], что к концуXIX–началу XX века в математике появились вычислительные задачи, для которых математики не могли предложить методов решения. Неудачные попытки математиков решить ту или иную задачу привели к мысли о том, что может и не существовать метода их решения. Но, чтобы доказать подобного рода утверждения, необходимо иметь математическое определение метода, то есть для интуитивного понятия вычислимости задачи необходим его формальный эквивалент. И, к чести математиков, в XX веке были сформулированы математические понятия (А. Черч и С.К. Клин ввели понятие ג– определимой функции, в1934 году К. Гедель рассмотрел класс функций, названных общерекурсивными, в 1943 году был предложен формальный эквивалент понятия вычислимой функции в виде модели Поста и др.), раскрывающие сущность интуитивных представлений о том, что такое метод (алгоритм) решения той или иной задачи, что такое вычислимая функция. Эти понятия явились основой научного направления, получившего название теории алгоритмов, которой посвящены работы [ и др.]

Важность теории алгоритмов вытекает не только из общенаучных проблем развития математики, но также из практических задач информационного общества, использующего информационные технологии в различных сферах и нуждающегося в адекватном представлении о возможностях электронных вычислительных машин. Нюанс заключён в том, что язык программирования (алгоритмический язык) должен соответствовать конкретной модели вычислений, реализуемой по-разному в машине Тьюринга, РАМ - машине (равнодоступная адресная машина), РАСП-машине (равнодоступная адресная машина с хранимой программой) или какой-либо другой. Прослеживается тесная связь между теоретическими моделями вычислений и реальным программированием. Так, например, ג – исчисление Черча является прообразом функционального программирования, реализованного в языке ЛСИП, а модель Поста содержит идеи, использованные в языках типа Фортран и Алгол. Методы логического программирования прослеживаются в версиях языка Пролог [ 1].

Языки программирования имеет свою историю, представляемую их поколениями (англ. generation programming language, GL).

К первому поколению (1GL) относятся языки программирования на уровне машинных кодов, в частности, языки ассемблера. Они называются машинными языками программирования.

Языки программирования второго поколения (2GL) были созданы для того, чтобы облегчить тяжёлую работу программистов, перейдя в языковых конструкциях от машинных кодов конкретной ЭВМ к выражениям, приближенным к тем, которыми мыслит программист. Эти языки появились в 1950-е годы, например, такие языки как Фортран и Алгол, которые называются машина ориентированными языками программирования.

Отличительной чертой языков программирования третьего поколения (3GL) стала их независимость от аппаратной платформы — это машинно-независимые языки программирования. Перевод алгоритмического языка программы на машинный код осуществляют трансляторы.

В период с 1970-х по начало 1990-х годов в развитие идей языков предыдущего поколения стали создаваться языки четвёртого поколения (4GL), ориентированные на специализированные области применения и оперирующие конкретными понятиями узкой предметной области. В эти языки встраиваются мощные языковые конструкции – операторы, описать такую функциональность, для реализации которой на языках младших поколений потребовались тысячи строк машинного кода. Языки этого поколения предназначены для реализации крупных программных проектов.

В середине 1990-х годов появились языки программирования пятого поколения (5GL), к которым относятся системы создания прикладных программ с помощью визуальных средств разработки, без знания программирования как такового.

Языки программирования с третьего по пятое поколения называются языками программирования высокого уровня.

Языки программирования используются для реализации программирования как процесса в виде упорядоченных действий, завершающегося необходимой программой. Любое действие всегда связано с чем-то или направлено на что-то. Исходя из этого, примем в общем виде (программирование может быть связано не только с компьютером, но и многим другим) следующее определение:

**Программирование** (греч. programma– объявление, распоряжение) – подготовка чего-то для действия на нём с чем-либо (например, подготовка компьютера для решения на нём задач).

Характер программирования компьютера определяется видом программирования, которое может быть, по крайней мере, одним из следующих: императивным, процедурным, декларативным, функциональным, объектно- ориентированным, логическим, визуальным [ ] или каким-либо иным.

**Императивное программирование** (лат.imperativus–повелительный) – предписание компьютеру выполнять при решении задач заданную последовательность команд.

**Процедурное программирование** – (франц. procedureот лат. procedo- продвигаюсь) – императивное программирование, при котором последовательно выполняемые команды (операторы) собраны в комплексы (подпрограммы) с помощью механизмов самого языка программирования.

**Декларативное программирование** (лат. declaration- объявление) – описание исходных данных задачи и ожидаемых результатов, автоматически порождаемое компьютерную программу для преобразования первого во второе, т.е. получения искомого результата. Возможны два варианта декларативного программирования: логический и функциональный. При логическом декларативном программировании используется формальный логический язык (пример – язык программирования ПРОЛОГ). При функциональном процедурном программировании используются соотношения между исходными данными и конечными результатами (пример – язык программирования ЛИСП).

**Функциональное программирование**–программирование, в котором процесс вычисления трактуется как вычисление значений функций в математическом их понимании (в отличие от функций как подпрограмм в процедурном программировании). К функциональному программированию относятся языки программирования ML, Miranda, Haskell, Cleanи др. [2 ].

**Объектно-ориентированное программирование**–программирование, основанное на использовании объектов и классов и учёте их взаимодействий [ 3].

**Логическое программирование** (греч. logike–наука о способах доказательств и опровержений. Основателем считается Аристотель) – программирование, основанное на автоматическом доказательстве теорем или, иначе, получении информации, являющейся результатом изучения фактов [4 ].

**Визуальное программирование** (лат. visuali - зрительный)–создание компьютерной программы путём манипулирования графическими объектами вместо написания её текста.

К визуальному программированию относятся языки программирования: Borland Delphi, C++ Buider, Дракон-схемы [5 ].

Следует принципиально заметить, что чётких и однозначно разделяющих границ между видами программирования на самом деле не существует. Однако идентифицирующие признаки, позволяющие отличить один вид программирования от всех остальных, выше сформулированы.

В итоге: в данном случае языки программирования подразделены, что не противоречит известным публикациям, на императивные, процедурные, декларативные, функциональные, объектно-ориентированные, графические, визуальные и, возможно, иные другие, соответствующие, по принадлежности, определённому поколению из множества GL = (1GL, 2GL, 3GL, 4GL, 5GL, …), где многоточие означает последующие поколения языков программирования.

Языки программирования логически замыкают информационно - технологическую доминанту, которая в пределах рассматриваемой базовой морфологической модели по существу является субстанцией, поглощающей интеллектуальную доминанту архитектуры АСОИУ. Рассмотрим интеллектуальную доминанту поближе.

### **8.1.6.2. Интеллектуальная доминанта**

Здесь:

**Интеллект** (лат.intelkectus– латинский перевод др.-греч. понятия   
“nus”)– мысль, ум, разум.

**Интеллектуальная**– способная мыслить, понимать, размышлять.

Информационно-технологическая доминанта, рассмотренная выше, создаётся человеком и для человека. Парадоксальным образом человек, создавая что-то сам для себя, образует петлю [1 ], которая, размножаясь, превращается в ячейки, порождающие сетки, которые, наконец, сливаются, в сеть, становящейся паутиной, пленяющей (сильнее, порабощающей) человека (в широком смысле – людей и, более того, человечество). Метафора, близкая к действительности.

Паутина – это феномен, окутывающий (распространяющийся вширь) и пронизывающий пространство и всё его содержимое (проникающий внутрь). Процесс развивается по линии: бинарный – дуальный – экзистенциональный.

Здесь:

**Бинарный** (лат. binaries – двойной) – двухкомпонентный, состоящий из двух частей, соподчинённых одна другой.

**Дуальный** (лат. dualis - двойственный) – двуначальный, признающий равноправие двух начал: духа и материи, идеального и материального, субъективного и объективного, виртуального и реального, одной и другой частей целого.

**Экзистенциональный** (лат. exsistentia - существование) – существование как нерасчленённая целостность двух начал: духа и материи, идеального и материального, субъективного и объективного, виртуального и реального, одной и другой частей целого.

Сначала возникли человеко-машинные системы, воплощающие бинарные отношения, при которых машина при человеке, т.е. подчинённая ему, выполняет чисто утилитарные функции (лат. utilitas– польза, выгода). В человеко-машинной системе человек и машина в известном смысле рассматривались и изучались независимо друг от друга.

Затем появились автоматизированные системы, в которых автоматическая и человеческая подсистемы приобрели в известном смысле самостоятельность, т.е. могли выполнять свои функции независимо друг от друга, отвечающие дуальному характеру отношений.

Наконец, настало время обоюдного центростремительного с ускорением приближения машины к человеку, с одной стороны, и человека к машине, - с другой, с неизбежным взаимопроникновением одного в другое и образованием симбиоза с экзистенциональным видом отношений в нём человека и машины. Развитие идёт по двум линиям: разработка искусственного интеллекта (назовём это условно созданием киберчеловека) и разработка искусственных органов для человека, и имплантация (вживление) в него технических устройств – чипов (создание человекокибера). Когда-то в перспективе может образоваться компания “человекокибер - киберчеловек” с сейчас труднопредсказуемыми последствиями. Здесь:

**Симбиоз** (греч. symbiosis - сожительство) – форма тесного существования двух организмов разных видов или объектов разной природы.

**Кибер**(англ. cyber) – слово-приставка, означающее связь с компьютером.

Заметим, что интеллект имеет, по крайней мере, две формы проявления. Во-первых, что-то созданное человеком и отчуждённое от него, ставшее достоянием кого-то другого, является носителем интеллекта его создателя (назовём это пассивным интеллектом). Во-вторых, кто-то, т.е. человек, пользующийся чем-то и, тем более, находящийся с ним в экзистенциональных отношениях, проявляет непосредственно свой интеллект (активный интеллект).

Таким образом, информационно-технологическая доминанта является носителем интеллекта её разработчиков (пассивного интеллекта), а пользователи, включённые в доминанту, привносят в неё свой активный интеллект непосредственно. Но разработчик чего-то, в свою очередь, является пользователем информации от источника, с которым он находится в соответствующих отношениях. Проявляется эффект петли, отмеченный выше, и тогда в общем случае:

**Интеллектуальная доминанта** – это пользователи, включённые в информационно-технологическую доминанту на условиях бинарности, дуальности или экзистенциональности.

Если пользователями являются разработчики автоматизированных информационных систем как центрального звена АСОИУ, то сфера их профессиональной деятельности может быть очерчена базовой функционально-структурной моделью архитектуры АСОИУ, рассматриваемой далееотдельно. Но, независимо от сферы профессиональной деятельности, трудовая активность пользователей должна быть скоординирована и направлена на достижение общих целей. Это может быть достигнуто ограничениями возможных действий и взаимодействий пользователей пределами определённой организационной структуры (оргструктуры). Поведение пользователей в условиях организации может быть описано базовой моделью поведения пользователя архитектуры АСОИУ, рассматриваемой далее отдельно.

Для эффективной трудовой деятельности пользователей они должны быть обеспечены персональными автоматизированными рабочими местами (АРМ) и для коллегиального обсуждения проблемных ситуаций должно иметься коллективное рабочее место – ситуационный центр (СЦ). При этом комфортабельные условия жизнедеятельности пользователей и функционирования техники обеспечиваются инженерно-строительным комплексом (ИСК) архитектуры АСОИУ.

Структура, автоматизированное рабочее место, ситуационный центр и инженерно-строительный комплекс интеллектуальной доминанты рассматриваются здесь далее. Но прежде ещё о пользователях.

### **8.1.6.2.1. Пользователи**

Любой человек постоянно нуждается в источнике информации и её приобретении и несть им числа и имя им легион. За короткое время информатизация на основе массового внедрения средств вычислительной техники создала виртуальное пространство, в котором нет дистанционного разрыва источник - потребитель информации, обеспечив информационный обмен между источниками и потребителями в режиме реального времени (режим on-line), в общем случае, инвариантно к их фактическому местонахождению.

Феномен информационной зависимости позволяет идентифицировать каждого человека как пользователя информации (в том числе, и как пользователя автоматизированной информационной системы, которая является источником информации), соотнеся ему формально понятие “пользователь”, что и было сделано выше. Там же определены классы пользователей:

Лица, принимающие решения, - ЛПР,

Лица, готовящие решения, - ЛГР,

Лица, участвующие в принятии коллегиальных решений, - КЛГ,

Лица, обслуживающие принятие коллегиальных решений - реализующие сценарий, - ЛРС,

Лица, исполняющие решения, - ЛИР,

Лица, готовящие информацию, - ЛГИ,

Лица, выполняющие повседневную работу, - ЛВР,

Лица, пользующиеся информационными услугами, - ЛПИ,

Лица, обеспечивающие функционирование системы, ЛОФ,

Лица, осуществившие создание системы и внедрение её в эксплуатацию, - ЛОВ.

Каждый из выделенных классов является функционалом.

Здесь:

**Функционал** (лат. function – исполнение) – множество функций, исполняемых кем-то, или реализуемых чем-то (например, компьютером).

Конкретный человек может выполнять один из этих функционалов или несколько, в том числе, и все. Кратко охарактеризуем каждый функционал, имея в виду, что функционалы находятся в связях и отношениях, задаваемых информационно-технологической доминантой и, конкретно, присутствующей в ней дихотомией “управляющий орган (или объект)- объект управления)”.

**Лицо, принимающее решение**. Это – пользователь, наделённый полномочиями единолично принимать окончательные управленческие решения и отвечать за последствия их реализации. Он является единственным лицом любой организации, правомерно имеющим такие полномочия. Это может быть Президент страны, глава корпорации, индивидуальный предприниматель, т.е. первое лицо организации. Во многих случаях, кроме обыденных, принятие управленческого решения является выбором из множества альтернатив, предлагаемых лицами, готовящими решения.

Здесь:

**Организация** (франц. organization, отпозднелат. organize – сообщаю стройный вид, устраиваю) – упорядоченность и согласованность частей целого.

Организация, прежде всего, но не всегда, представляется коллективом, в котором люди совместными усилиями достигают общие цели (цели организации).

**Альтернатива** (франц. alternative, от лат. alter– один из двух) – необходимость выбора одного из двух или многих чего-то (решений, направлений, вариантов и т.п.).

**Лицо, готовящее решение**. Это – пользователь, который в пределах личной компетенции формулирует для ЛПР предложения по конкретному управленческому решению. Он является не единственным ЛГР, что позволяет рассматривать и аргументировать будущее управленческое решение с различных точек зрения и предлагать два или более конкурирующих вариантов, которые образуют для ЛПР множество альтернатив выбора.

Здесь:

**Компетенция** (лат. compete – соответствую, подхожу) – знания, умения и опыт кого-то в конкретной области.

К лицам, готовящим решения, следует отнести сотрудников, например, Администрации Президента. То же самое имеет место при первых лицах других организаций или объектов.

В сложных случаях, когда возникают проблемные ситуации, для выработки управленческого решения привлекается коллегия – КЛГ.

**Коллегия**. Это – ситуативный, т.е. обусловленный текущей ситуацией, коллективный пользователь, состоящий из высокопрофессиональных специалистов в узкой области знаний – экспертов.

Здесь:

**Эксперт** (франц. expertiseот лат. expertus–опытный) – человек, обладающий уникальными, только ему присущими, познаниями в каком-либо конкретном вопросе.

Формы и способы формирования коллегии разнообразны, но для совместной работы экспертов им необходимо предоставлять нужную информацию как на средства отображения индивидуального, так и на средства отображения коллективного пользования. Эта информация предварительно структурируется в соответствии со сценарием, отражающем специфику проблемной ситуации. Сценарий предписывает последовательность вывода информации на средства отображения в темпе её использования коллегией. Вывод информации для коллегии осуществляет лицо, обслуживающее выработку коллегиальных решений, - лицо, реализующее сценарий (ЛРС).

**Лицо, обслуживающие принятие коллегиальных решений**(реализующее сценарий -ЛРС). Это – пользователь, ответственный за разработку сценария, соответствующего проблемной ситуации, и его реализацию с возможной коррекцией в процессе выработки альтернатив коллегиального управленческого решения.

ЛПР, ЛГР, КЛГ и ЛРС принадлежат управляющему органу.

Принятые управленческие решения персонифицированы, т.е. предназначены для реализации конкретными лицами, исполняющими решения.

**Лицо, исполняющее решение**. Это – пользователь на объекте управления, который обязан реализовать принятое управленческое решение.

Лицом, исполняющим решение, на объекте управления является его первое лицо, т.е. руководитель и его аппарат (заместители, помощники и т.д.).

Любое управленческое решение в той или иной степени связано с переводом объекта управления из текущего состояния в требуемое. Информация о прошлом, текущем и будущем состоянии объекта не всегда, точнее, во многих случаях не может формироваться автоматически без участия человека – лица, готовящего информацию – ЛГИ, который является гарантом её правомерности и достоверности.

**Лицо, готовящее информацию**. Это – пользователь, который обязан в пределах предоставленных ему полномочий готовить в установленном виде информацию о состоянии объекта управления и передавать её в информационно-технологическую доминанту.

Функционирование объекта управления, а также в равной степени и управляющего органа сопровождается решением множества текущих задач, касающихся финансов, снабжения, социального обеспечения, юридических аспектов, кадров, подготовки и переподготовки персонала, модернизации, развития и т.п., которые являются рутинными, т.е. решаемыми одними и теми же известными способами, но для различных исходных данных. Эти задачи решаются специалистами, относящимся к лицам, выполняющим повседневную работу, - ЛВР.

**Лицо, выполняющее повседневную работу**. Это – пользователь, обязанный в пределах своей компетенции решать рутинные задачи в обеспечение функционирования объекта или органа.

Функционирование объекта управления и управляющего органа как целого, т.е. системы в широком смысле - АСОИУ, протекает в условиях активного информационного взаимодействия с окружающей средой, когда система приобретает из среды доступную и необходимую для себя информацию и выдаёт в среду в допустимых пределах свою информацию, нужную для лиц, пользующихся информационными услугами системы.

**Лицо, пользующееся информационными услугами**. Это – сторонний, т.е. за пределами системы, пользователь, которому доступны информационные услуги системы.

Функционирование автоматизированной системы обработки информации и управления сопровождается воздействием на неё множества внутренних и внешних возмущающих воздействий, большая часть которых не контролируема и не предсказуема, кроме того, некоторые из них имеют преднамеренный целенаправленный разрушающий характер. Это представляет работоспособности системы серьёзную угрозу, которую способны нейтрализовать лица, обеспечивающие её функционирование - ЛОФ.

**Лицо, обеспечивающее функционирование**. Это – пользователь, который обязан в пределах свой зоны ответственности обеспечить работоспособность системы.

**Лицо, обеспечившее создание системы и внедрение её в эксплуатацию**. Это – Главный конструктор системы и в его лице многочисленный коллектив разработчиков.

Заметим, что автоматизированная информационная система как центральное звено автоматизированной системы обработки информации и управления для пользователей классов ЛПР, ЛГР, ЛИР, ЛГИ, КЛГ, ЛОР, ЛВР, ЛПИ является орудием труда, для пользователей классов ЛОФ и ЛОВ – это предмет труда.

Пользователь – это натуральный живой человек, являющийся воплощением и носителем естественного интеллекта, который определяет образ действия человека и в нём же проявляется. Следует отметить, что потенциальные пользователи, как интеллектуальная элита рассматриваемой доминанты, если воспользоваться теорией поколений [ ], относятся, частично, к поколению **X**(родившихся в 1963 – 1983 гг.) и, в большей части, к поколению **Y**(родившихся в 1983 – 2003 гг.).

Поколение **X**отличают фундаментальность образования, стремление к знаниям, прагматизм и техническая основательность.

Поколению **Y** присуще:

разностороннее образование, нередко, подкреплённое к 23 годам несколькими дипломами,

лабильность, динамичность и адаптивность к изменяющимся условиям,

свободное владение девайсами, Интернетом, активное пользование медиаресурсами, интерес к IT-технологиям,

стремление иметь интересную работу с гибким графиком пребывания в офисе и хорошую зарплату, а также достаточно свободного времени,

приверженность к комфорту, удобству и эстетике в отношении себя и вокруг.

С учётом этих особенностей потенциальных пользователей их совместная работа каждого на своём рабочем месте должна быть скоординирована и целенаправленна на достижение общих целей и это обеспечивается с помощью структуры, суть которой состоит в следующем.

### **8.1.6.2.2. Структуры**

В общем случае:

**Структура** (лат. structura – строение, расположение, порядок) – упорядоченное множество элементов и связей между ними.

Элементы и связи могут быть двух видов – физические и логические в различных сочетаниях. Если элементы и связи имеют физическую природу, то структура становится структурной схемой:

**Структурная схема** – это графическая модель, описывающая упорядоченное множество конструктивных элементов системы и физических связей между ними.

Если элементы и связи имеют логическую природу, то структура становится функциональной схемой:

**Функциональная схема** – это графическая модель, описывающая упорядоченное множество функций, реализуемых системой, и логических связей между ними.

Наконец, множество элементов может представлять группа людей. Тогда структура преобразуется в организацию:

**Организация** (франц. organizationот позднелат. organize–сообщаю стройный вид, устраиваю) – объединение людей, действующих целенаправленно, т.е. коллективно.

Во многих других случаях организация рассматривается как система, обладающая внутренней структурой, содержащей элементы различной физической природы и многообразные связи: физические, технологические, экономические, правовые и, в том числе, человеческие взаимодействия. В таких случаях структура и организация в широком смысле как понятия являются синонимами.

Особенностью коллектива, как самостоятельной организации или составной части организации в расширенном толковании, является то, что входящие в него люди находятся между собой в определённых отношениях, основой которых является моральная соподчинённость, простирающаяся от полной независимости до предельной покорности.

Здесь:

**Мораль** (лат. moralis– нравственный) – способ регулирования действий человека с помощью норм (что можно и нужно) и правил (что при этом следует соблюдать).

Целенаправленная деятельность коллектива обеспечивается его организационной структурой или, короче, оргструктурой. Заметим, что коллектив может меняться частично или полностью, а оргструктура – при этом оставаться неизменной. Тогда:

**Оргструктура** – это графическая модель, описывающая распределение норм и правил внутри организации, обязательных для её членов (реальных для существующей и потенциальных для создаваемой организации).

Организацию в расширенном толковании представим инфограммой, приведённой на рис. 8.6.2.1.

Здесь:

**Инфограмма** (лат. information – разъяснение, изложение + греч. gramma– черта, буква, написание, часть сложного слова, означающая графическое изображение) – графическое представление информации о чём-либо для максимально быстрого усвоения его смысла.

Из инфограммы на рис. 8.6.2.1 следует, что оргструктура является центральным элементом организации, интегрирующим в ней в условиях окружающей среды в целое производство, ресурсы, финансы, потребителей и производителей. В отличие от организации, которая по определению и, по существу, должна быть стабильным образованием, окружающая среда фактически характеризуется динамичностью и слабой предсказуемостью и может резко и кардинально меняться.

Поэтому необходимым условием жизнестойкости и эффективности организации является чувствительность её оргструктуры к внешней среде с тем, чтобы своевременно адаптироваться к изменившимся условиям функционирования организации. Отсюда следует принципиальная уникальность реальных оргструктур и интегрированных ими организаций. Тем не менее, их основу образуют типовые организационные структуры, к которым можно отнести, в частности, следующие наиболее известные оргструктуры []:

линейные,

функциональные,

линейно-функциональные,

линейно-функциональные штабные,

матричные,

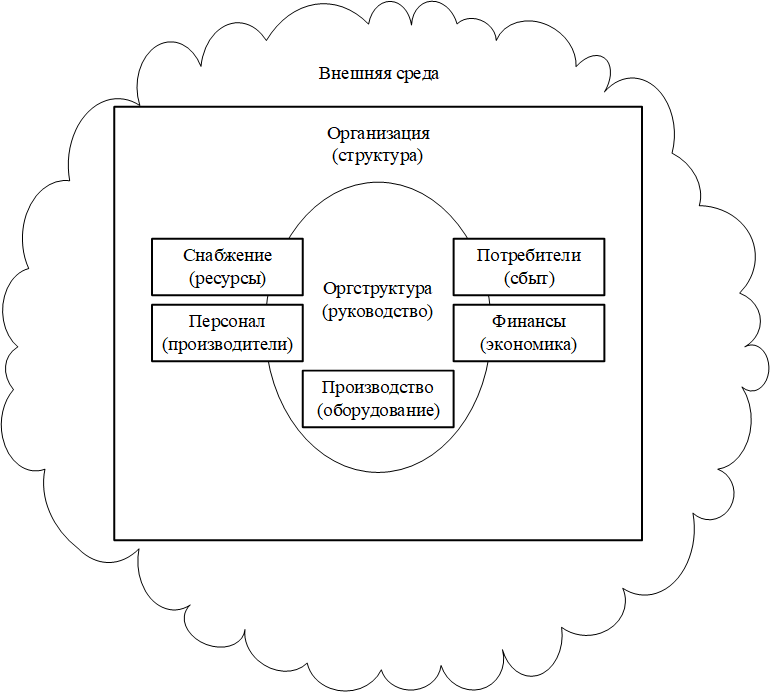


Рис. 8.6.2.1. Инфограмма организации

дивизиональные,

сетевые.

Кратко охарактеризуем эти оргструктуры.

**Линейная оргструктура**. Графическая модель линейной оргструктуры представлена на рис. 8.6.2.2. Это простейший вид оргструктуры, придуманный человеком ещё в древние времена и применяемый до сих пор. Основные принципы линейной организации управления – это централизация власти и единоначалие. Каждое звено оргструктуры предназначено для единоличного руководителя, наделённого всем объёмом полномочий по отношению к руководителям подчинённых ему звеньев. Распоряжения в этом случае передаются сверху вниз, последовательно от уровня к уровню. У каждого работника есть только один начальник, но у каждого руководителя (начальника) – несколько подчинённых. Официально разрешенных горизонтальных связей в данной оргструктуре не существует.

Основные достоинства линейной оргструктуры – простота реализации на практике и однозначная личная ответственность каждого руководителя в пределах его компетенции.

Здесь:

**Компетенция** (лат. kompetentia– принадлежность по праву) – круг полномочий или познаний и умений кого- то.

Недостатком линейной оргструктуры является её плохая адаптация (приспособляемость) к постоянно и непредсказуемо меняющейся внешней среды. Также высоки требования к квалификации руководителей и особенно её верх-

него звена. Эти недостатки являются первопричиной перехода к функциональной и другим организационным структурам.

**Функциональная оргструктура**. Графическая модель функциональной оргструктуры представлена на рис. 8.6.2.3. Здесь выделяются функциональные звенья (снабжения, производства, финансирования, сбыта и т.п.). Это позволяет использовать высококвалифицированных узкого профиля специалистов, ориентированных на соответствующую конкретную функцию, что является необходимым условием принятия качественного управленческого решения в её пределах. Выполнение решений функционального звена является обязательным для подчиненных звеньев.

Основное достоинство функциональной оргструктуры – отсутствие критической потребности в высококвалифицированных специалистах широкого профиля, недостаток – вследствие нарушения линейности подчинения (подчинённость данного звена нескольким звеньям вышестоящего уровня) возникает конфликт интересов. Этот недостаток удаётся ослабить в линейно - функциональной оргструктуре.

**Линейно-функциональная оргструктура**. Графическая модель линейно- функциональной оргструктуры представлена на рис. 8.6.2.4. Эта оргструктура обеспечивает горизонтальное разделение функциональных звеньев на всех уровнях, сохраняя в целом линейность соподчинения. Интеграция управленческих решений,

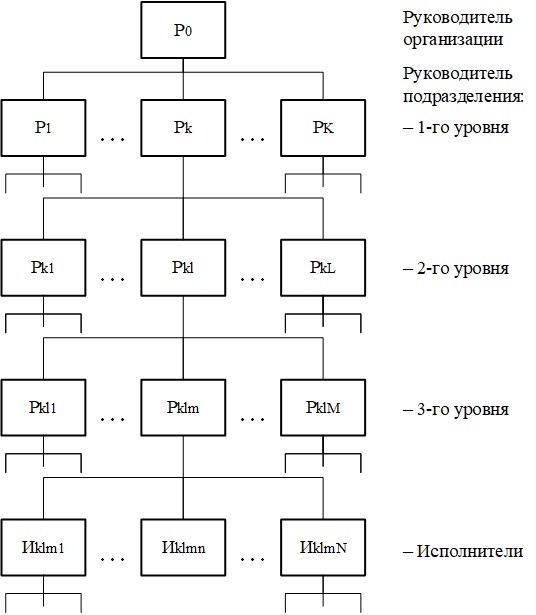
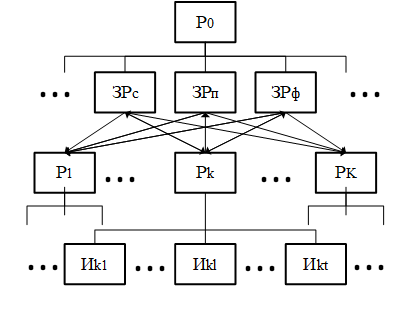


Рис. 8.6.2.2. Линейная оргструктура управления

Обозначения:

– звено оргструктуры или оргячейки

– вертикальные связи (отношения)



– Руководитель организации

– Зам. Руководителя по снабжению, С, производству, П, финансированию, Ф и др.

– Руководитель подразделений

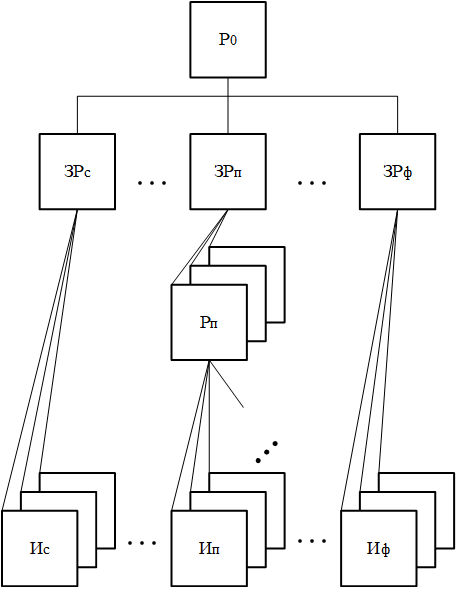
– Исполнители

Рис. 8.6.2.3. Функциональная оргструктура управления

Обозначения:

– звено оргструктуры или оргячейки

– вертикальные связи (отношения)



– Руководитель организации

– Зам. руководителя по снабжению, производству, финансам и др.

– Руководители производственных подразделений

– Исполнители

Рис. 8.6.2.4. Линейно-функциональная оргструктура управления

Обозначения:

 – звено оргструктуры или оргячейки

 – вертикальные связи (отношения)

соответствующих линейным вертикалям оргструктуры, происходит на верхнем уровне в звене руководителя организации.

Линейно-функциональная оргструктура обеспечила качественно новое разделение труда в управлении и более эффективные управленческие решения. Но при решении проблемных задач, обусловленных нестабильностью внешней среды, данная оргструктура становится не эффективной, так как мало приспособлена к стратегическому планированию. Для преодоления этой неприятности появились линейно-функциональные штабные структуры.

**Линейно-функциональная штабная оргструктура**. Графическая модель линейно-функциональной штабной оргструктуры приведена на рис. 8.6.2.5. Эта оргструктура включает специализированное звено – штаб и его звенья – секции, которые не наделены правами принятия решений и руководства, но помогают соответствую-

щему руководителю в выполнении функций и решении задач, прежде всего, функций и задач стратегического планирования и анализа. Основная цель стратегического планирования и анализа – создание потенциала для выживания организации в условиях изменяющейся внешней среды, порождающей неопределённости. Эта оргструктура обеспечивает возможность привлечения внешних консультантов и экспертов.

Основным недостатком линейно-функциональной штабной оргструктуры является недостаточно чёткое распределение ответственности, так как лица, принимающие решения, не участвуют в их реализации, а также чрезмерная централи-

зация управления, сосредотачивающаяся на верхнем звене оргструктуры.

В известной мере этот недостаток преодолевает матричная оргструктура.

**Матричная оргструктура**. Графическая модель матричной оргструктуры представлена на рис. 8.6.2.6. Главной особенностью матричной оргструктуры является наличие горизонтальных связей и множественная подчинённость, чаще всего двойная, как в данном случае на рисунке, - руководителю функционального звена и руководителю проекта. Матричной оргструктуре присуща децентрализация управлен-

ческих функций и снижение нагрузки и повышенных профессиональных требований к руководителям верхних уровней оргструктуры. Матричная оргструктура обладает большей гибкостью по сравнению с предыдущими оргструктурами и лучшей адаптацией к изменениям внешней среды (легче заменить проекты или продукты на другие и перестроить функциональные звенья).

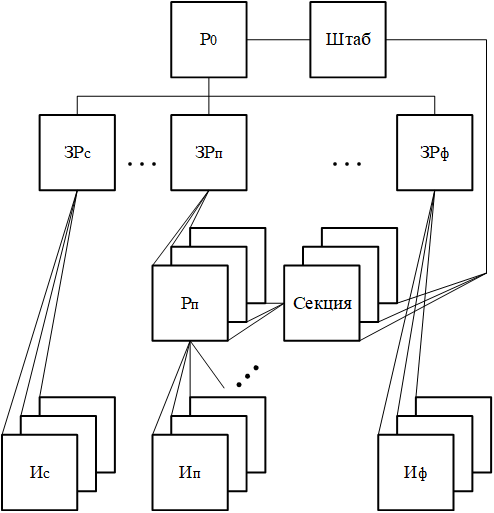


Рис.8.6.2.5. Линейно-функциональная штабная оргструктура управления

Обозначения:

 – звено оргструктуры или оргячейки

 – вертикальные связи (отношения)

Основной недостаток матричной оргструктуры таится в возможностях острых противоречий между сторонами матрицы (руководителями проектов - продуктов и функциональными руководителями), а также нарушении принципа единоначалия. И то и другое в значительной степени связано с индивидуальными качествами руководителей и культурой, и искусством человеческих отношений.

Эти недостатки в меру удаётся преодолеть в дивизиональной оргструктуре, сохранив при этом гибкость и адаптируемость её к непредсказуемости окружающей (внешней) среды.

**Дивизиональная оргструктура**. Дивизиональная (фр. division– деление, разделение) оргструктура интегрирует несколько дивизионов (автономных звеньев – подразделений), выделенных:

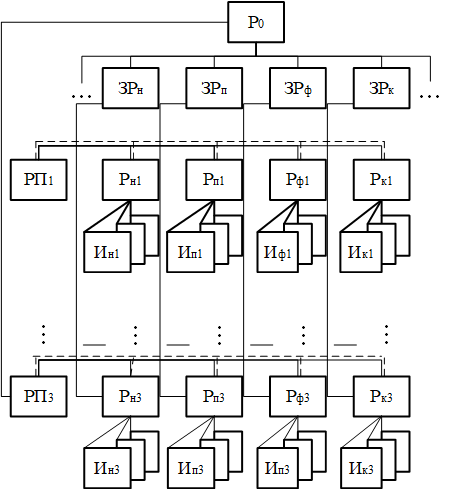
либо по виду выпускаемой продукции (например, электроника, одежда, канцелярские товары и т.п.),

либо по региональному признаку (выпуск одной и той же продукции в разных регионах),

либо по классам потребителей (армия, молодёжь, VIP– покупатели и т.п.).

Графическая модель дивизиональной оргструктуры представлена на рис. 8.6.2.7. Здесь дивизионы имеют полную оперативно-производственную самостоятельность и несут в полном объёме ответственность за эффективность своей деятельности. Преимуществом дивизиональной оргструктуры является оперативное реагирование на изменения, возникающие во внешней среде.

Основной недостаток дивизиональной оргструктуры проявляется в трудностях, связанных с согласованием интересов дивизионов, которые не настолько часты и остры, как в матричной оргструктуре.



Руководитель проекта 1

(или продукта 1),

Руководитель проекта 3

(или продукта 3),

– Руководитель организации

– Зам. Руководителя по науке (ИНОКР), производству П, финансированию Ф, кадрам К и т.п.

– Руководитель подразделения по науке Н (НИОКР), производству П, финансированию Ф, кадрам К

– Исполнители

…

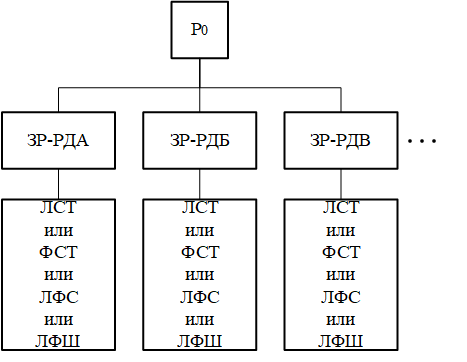
Рис. 8.6.2.6. Матричная оргструктура управления

Обозначения: – звено оргструктуры или оргячейки

– вертикальные связи (отношения)

– горизонтальные связи (отношения)

НИОКР – научно-исследовательская и опытно- конструкторская работа



– Руководитель организации

– Зам. руководителя – руководитель дивизиона, выпускающего продукцию А, Б, В…

Рис. 8.6.2.7. Дивизионная оргструктура управления

Обозначения:

ЛСТ – линейная оргструктура

ФСТ – функциональная структура

ЛФС – линейно-функциональная оргструктура

Объединение линейной и функциональной оргструктур в единую линейно-функциональную оргструктуру можно рассматривать, в частности, как прецедент, обусловивший слияние матричной и дивизиональной оргструктур и образования на этой основе сетевой оргструктуры.

Здесь:

**Прецедент** (лат. praccedens (praccedenis) – идущий впереди, предшествующий) – случай, имевший ранее место и служащий примером или оправданием для последующих случаев подобного рода.

**Сетевая оргструктура**. Графическая модель сетевой оргструктуры представлена на рис. 8.6.2.8. Будучи гибридной, сетевая оргструктура наследует преимущества исходных оргструктур: гибкость и адаптивность, присущие дивизиональной оргструктуре, и централизованное функциональное руководство, характерное для матричной оргструктуры.

Обязательные условия сети – это централизованное общее руководство, осуществляемое верхним звеном, и централизованные функциональные звенья по ключевым направлениям деятельности, распоряжения которых обязательны для всех дивизионов. Отсюда следует единый сетевой стиль, общий для сети бренд, одни и те же поставщики, один и тот же ассортимент продукции (товаров или услуг) и т.п., но в то же время в определённом смысле самостоятельность, позволяющая каждому дивизиону адаптироваться к своему окружению.

Наиболее эффективны территориально распределённые сетевые оргструктуры с единым стилем и брендом, что обеспечивает её узнаваемость не зависимо от того, где находятся её дивизионы.

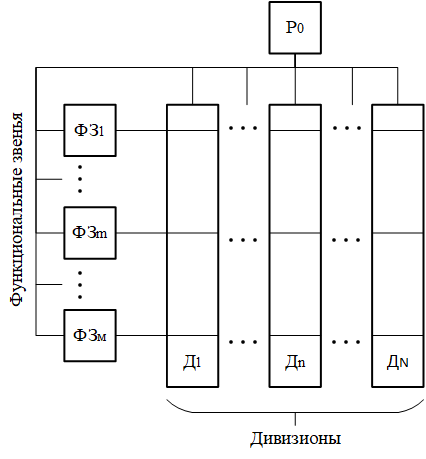
Рассмотренные оргструктуры наиболее известны и распространены, но далеко не исчерпывают всего разнообразия существующих и известных организационных структур. К тому же оргструктуры реальных организаций, как правило, являются комбинацией типовых структур. При этом каждая комбинация учитывает особенности, вытекающие из уникальности организации и её окружающей среды.

Однако, при всём при этом, не зависимо от того, является ли оргструктура типовой или реальной, общим для них является то, что в той или иной степени, но, как правило, до сих пор значительной, они активно используют административный контроль и осуществляют посредством него администрирование в организации. Это – во-первых.

Во-вторых, всем рассмотренным типовым и соответствующим им гипотетическим реальным организациям в той или иной степени присуща иерархия.

Здесь:

**Контроль** (фр. controle) – проверка, наблюдение соответствия чего-либо чему-либо.



– Руководитель организации

Рис. 8.6.2.8. Сетевая оргструктура управления

**Административный** (от лат. administratio –управлять, заведовать) - исполнимый формально, т.е. по форме, официально, по жёстко установленным правилам.

**Администрирование** (лат. administratio –управлять, заведовать) - управление формально, т.е. по форме, посредством приказов, команд, в общем, властных полномочий или, короче, власти.

Здесь:

**Власть** – способ управления поведением человека и людей волевыми усилиями без физического насилия (с помощью приказа, команды, порицания, поощрения, авторитета и т.п.).

**Административный контроль** – наблюдение соответствие чего-либо чему-либо по жестко установленным правилам.

**Иерархия** (греч. hierarchia ≤ hieros– священный + arche- власть) - расположение частей целого в порядке от высшего к низшему с соблюдением подчинения.

Администрирование и иерархия, возведённые в абсолют, порождают бюрократизм, под которым будем понимать следующее:

**Бюрократизм** или **бюрократия** (фр. bereau– коллегиальный орган (бюро, канцелярия) для выполнения какой-либо работы, чаще, руководящей + греч. kratos– власть, господство) – в любых условиях и при разных обстоятельствах сосредоточение (централизация) власти в руках меньшинства и её исключительное распределение между членами меньшинства.

Бюрократизм в полной мере присущ и является становым хребтом любой линейной оргструктуре, которую в силу этого принято называть иерархической или бюрократической оргструктурой. В меньшей, но, тем не менее, значительной степени, бюрократизм присущ и другим рассмотренным выше типовым оргструктурам. Бюрократизм нейтрализует свободу и личную инициативу, расширение круга обязанностей и ответственности любого человека, включенного в ту или иную из традиционных организационную оргструктуру, и это ухудшает её функционирование и, тем более, в турбулентных условиях внешней среды, сложность которой нарастает.

Этот фактор является решающим для интенсивного поиска если не оптимальных, то хотя бы рациональных, по сравнению с традиционными, оргструктур, адекватных современным условиям. Сейчас вектор поиска ориентирован на противопоставлении бюрократии эдхократии.

Здесь:

**Эдхократия** (**адхократия**) (лат. anhoc– для данного случая, к этому + греч. kratos– власть, господство) – рассосредоточение (децентрализация) власти между многими и её применение с учётом складывающихся условий и конкретных обстоятельств.

Эдхократия ориентирована на власть, основанную на свободном творчестве, компетенции и добровольном сотрудничестве, а не на позиции в иерархии, что характерно для бюрократии. Объективность отхода от бюрократии и приближения к эдхократии обусловлена реальностью общественного развития по вектору“ … - индустриализация – информатизация – концептуализация - …”, рассмотренному выше, и трансформацией традиционной социально - технологической среды жизнедеятельности человека в современную интеллектуальную информационно-технологическую среду, являющуюся здесь предметом рассмотрения.

Эдхократические задатки присутствуют в современных матричных, дивизиональных и сетевых оргструктурах. Поэтому уже сейчас их именуют эдхократическими оргструктурами (или органическими, или адаптивными). И эта тенденция набирает силу – рассматриваются интеллектуальные оргструктуры (переход от административного контроля к прямой ответственности исполнителей) [ ] и другие [ ].

Эдхократическая децентрализация направлена на создание мелких и непостоянных оргструктур с неформальным поведением и узкой специализацией [ ]. Глубина децентрализации может быть различной. С её увеличением центростремительные свойства оргструктуры ослабевают, превращая её сначала в формат для возможного общения взаимно заинтересованных лиц, а далее, делая их независимыми друг от друга.

Здесь:

**Формат** (франц. formatот лат. forma – вид, наружность) – что-то без четких формальных связей и отношений, распределяющих внутри права, обязанности, ответственность и т.п.

Иначе:

**Формат** – противоположность или антипод организации.

Но человек в интеллектуальной информационно – технологической среде сам по себе является эдхократом, наделённым полнотой власти над собой и полномочиями распоряжаться ею по своему усмотрению. Это, в частности, способствует естественному самоорганизующимся образом без принуждения формированию комьюнити, т.е.

**Комьюнити** (англ. community – сообщество) –сообщество или группа людей со схожими интересами, добровольно и инициативно общающихся между собой в виртуальном пространстве опосредованно через интернет.

**Интерес** – это предвестник цели.

И тогда комьюнити, как сообщество людей, увлечённых общей целью, — это мощное средство для её достижения и, особенно, в идущем на смену информационному концептуальном обществе, где основное место уготовано человеческому таланту.

В обозримой перспективе изменится и рабочее (офисное) пространство оргструктуры. Из индустриального понимания конторы, где служащие трудятся в рабочие часы, с кабинетами начальников и открытой рабочей площадкой для сотрудников (опен спейсом), офис превратится в арендованное место (своего рода коворкинг) для совместной работы разных специалистов на время обсуждения полученных результатов, обмена мнениями по сложным вопросам и направлениям дальнейшей работы.

Здесь:

**Коворкинг** (англ. co-working – совместная работа)– совместная работа фрилансеров в едином месте на добровольных началах.

**Фрилансер** (англ. freelancer – свободный работник) – специалист с творческими наклонностями, способный самостоятельно выполнять работу в комфортных условиях, свободных от официальных требований (фиксированный режим работы и отдыха, корпоративный стандарт, канцелярские условия и т.п.).

**Опен спейс** (англ. openspace– открытая площадка, свободная планировка) –офисная площадка с рабочими местами, разделенными легкими перегородками, вместо кабинетно - коридорного типа.

Таким образом, организационная структура имеет широкий диапазон возможных реализаций: от жестко бюрократической линейной до крайне свободной, лишенной какого-либо администрирования и построенной на добровольном творческом отношении к труду, стимулируемому конечным результатом. Однако, ни одна типовая оргструктура в чистом виде не применяется и реальная оргструктура – это сочетание нескольких типов, адекватное уникальности конкретной организации в неповторимых условиях её внешнего окружения. И главное – любая реальная оргструктура упорядочивает управленческий персонал, численность которого порой бывает внушительной, соотнося каждого управленца определённой ячейке оргструктуры, наделённой функциями, задачами, ответственностью, обязанностями и правами, тем самым обеспечивая слаженную коллективную деятельность по достижению общих целей организации.

Тем не менее, при всём многообразии и неповторимости реальных оргструктур общим для них является то, что любая оргструктура должна обеспечить каждого управленца рабочим местом, позволяющим эффективно использовать современные информационные технологии, или, иначе, автоматизированным рабочим местом. Оказавшись на автоматизированном рабочем месте, управленец становится пользователем информации, принадлежащим одному (или попеременно нескольким) из ранее выделенных классов пользователей.

Рассмотрим далее подробнее автоматизированные рабочие места пользователей.

**Контрольные вопросы для самопроверки знания**

**лекционного материала**

**КВ №388.** Дать лекционные определения понятий “программа”, “программное обеспечение” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Физический смысл общего программного обеспечения и специального программного обеспечения. На основе лекционного материала представить состав ПО ГАС “Контур” в виде графической модели (декомпозиционной схемы) и её краткая характеристика (состав элементов, их назначение и реализующие элементы программные средства).

**КВ.№389.** Дать лекционные определения понятий “программа”, “программное обеспечение” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Физический смысл общего программного обеспечения и специального программного обеспечения. На основе лекционного материала представить состав ПО ГАС “Выборы” в виде графической модели (декомпозиционной схемы) и её краткая характеристика (состав элементов, их назначение и реализующие элементы программные средства).

**КВ №390.** Дать лекционные определения понятий“язык программирования”, “алгоритмический язык”, “алгоритм”и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Поколения языков программирования и их краткая характеристика (физический смысл). Поколения языков программирования, адекватные использованным при создании ГАС “Контур”.

**КВ №391.** Дать лекционное определение понятия “программирование” и довести воспроизведение определения до автоматизма, т.е. быстро и правильно по памяти. Виды программирования и их краткая характеристика (физический смысл). Виды программирования, адекватные использованным при создании ГАС “Выборы”.

**КВ №392.** Дать лекционные определения понятий“, “бинарный”, “дуальный”, “экзистенциональный”, “интеллектуальный”, “интеллектуальная доминанта” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Охарактеризовать в исторической ретроспективе СОИ, АСОИ, АИС. АСОИУ, ГАС “Контур” и ГАС “Выборы” в контексте бинарности, дуальности, экзистенциональности.

**КВ №393.** Дать лекционные определения понятий “пользователь”, “функционал”, “организация”, “компетенция”, “эксперт”, “коллегия” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Классы пользователь и краткая характеристика их функционалов. Состав пользователей ГАС “Контур” и ГАС “Выборы”: что общего и какие их отличительные признаки.

**КВ №394.** Дать лекционные определения понятий “структура”, “структурная схема”, “функциональная схема”, “организация”, “оргструктура”, “функционал”, “инфограмма” и привести из лекции инфограмму организации. Довести воспроизведение определений и инфограммы до автоматизма, т.е. быстро и правильно по памяти. Краткая характеристика инфограммы организации. Конкретные структуры и оргструктуры, рассмотренные на лекциях и их краткая характеристика (графические модели, функционалы).

**КВ №395.** Дать лекционные определения понятий “система”, “объект”, “свойство” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Сформулировать физический смысл предпосылок создания систем, их назначения и объектов автоматизации (предметных областей). Провести сопоставительный анализ ГАС “Контур” и ГАС “Выборы”, перечислив 10 или более одинаковых свойств систем и 10 или более признаков, по которым системы отличаются друг от друга. -

**КВ №396.** Дать лекционные определения понятий “структура”, “структурная схема”, “функциональная схема”, “организация”, “оргструктура”, “функционал” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Состав типовых оргструктур и их краткая характеристика. Оргструктуры, использованные в ГАС “Контур” и ГАС “Выборы”, их графические модели и функционалы.

**КВ №397.** Дать лекционные определения понятий “бюрократия”, “эдхократия (адхократия)”, “формат”, “организация”, “комьюнити”, “коворкинг”, “фрилансер” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Сравнительный анализ бюрократизма и эдхократизма (преимущества и недостатки). Необходимость, возможности и целесообразность преодоление власти меньшинства над большинством.