

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ПРИРОДООБУСТРОЙСТВА



TEMPUS-SWARP-ICT 21051

Разработка учебных программ по управлению
почвенными ресурсами с использованием
информационно-коммуникационных технологий

Л. М. Рекс

**СИСТЕМНЫЕ ИССЛЕДОВАНИЯ И ИНФОРМАТИКА
ДЕЯТЕЛЬНО – ТЕХНО – ПРИРОДНЫХ СИСТЕМ**



**УЧЕБНОЕ ПОСОБИЕ
ЧАСТЬ 1.**

Москва 2004

**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
ДЕПАРТАМЕНТ КАДРОВОЙ ПОЛИТИКИ И ОБРАЗОВАНИЯ**

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ПРИРОДООБУСТРОЙСТВА



TEMPUS-SWARP-ICT 21051

Разработка учебных программ по управлению
почвенными ресурсами с использованием
информационно-коммуникационных технологий

Л. М. Рекс

**СИСТЕМНЫЕ ИССЛЕДОВАНИЯ И ИНФОРМАТИКА
ДЕЯТЕЛЬНО – ТЕХНО – ПРИРОДНЫХ СИСТЕМ**

**УЧЕБНОЕ ПОСОБИЕ
ЧАСТЬ 1.**

Допущено УМО по образованию в области
природообустройства и водопользования
в качестве учебного пособия
для студентов высших учебных заведений,
обучающихся по направлению подготовки
"Природообустройство"656400,
"Водные ресурсы и водопользование"656800

Москва 2004

ББК 14.2.5 14.4
УДК 504: 658: 681
Р 36

Рецензенты:

Доцент кафедры «Вычислительной техники и прикладной математики» МГУП

А. В. Ильинко

Главный научный сотрудник ВНИИГИМ, д.т.н.

П. Добрачев

Ю.

Рекс Л.М.

Р 36 «СИСТЕМНЫЕ ИССЛЕДОВАНИЯ И ИНФОРМАТИКА
ДЕЯТЕЛЬНО – ТЕХНО – ПРИРОДНЫХ СИСТЕМ» –Часть 1: Учебное пособие-
М., МГУП, 2004. - с.218.
ISBN 5–89231–143-0

В проблеме развития мыследеятельности кроме гениальности мыслителя представляется целесообразным иметь как бы помощника в виде "карты знаний". Она представляла бы мыслителю, как бы с "птичьего полета" возможность осматривать имеющиеся "придумки" в сжатом, выделенном, формализовано обозначенном виде. Наподобие карт географических, почвенных, геологических и т.д., т.е. некоторых моделях действительности и абстракций. Нам представляется, что методологической и методической основой может являться концепция пространства категориально-понятийных матриц, связанных с созданием различных продуктов (результатов деятельности).

Разработанные образцы таблиц и программ позволяют понимать технологию.

Пособие знакомит студентов со стандартным программным обеспечением (HTML), учит студентов применять компьютерные методы в познании нового знания.

Пособие предназначено для магистров, обучающихся по специальностям экологического профиля и природообустройства.

ISBN 5–89231–143-0

© Рекс Л.М.

© Tempus-SWARP-ICT 21051, 2003

Суть дела состоит в следующем:

1. Рассматривается макро уровень, т.е. природная среда ,на которой имеется множество технических объектов разного назначения (техно-природная система - ТПС). В рамках этой техно-природной системы протекают две деятельности природная и человеческая, т. е. в конечном представлении мы имеем дело с деятельно-техно-природными системами (ДТПС).

2. Человеческая деятельность состоит из 9 компонентов: человек, техника (инженерные системы), среда (в широком смысле: политическая, социальная, экономическая, техническая и природная, слагающаяся из климатических, гидрологических, гидрогеологических, инженерно-геологических и почвенных условий), ресурсы (в широком смысле), информация, модели (в широком смысле: физические и абстрактные), время, управление и продукт (в широком смысле: политический, социальный, экономический, технический, природный - измененный и интеллектуальный).

3. Следующим этапом строится категорийно - понятийное пространство (КПП) или матрица, т. е. таблица по следующему правилу: 8 компонентов (человек, техника, среда, ресурсы, информация, модели, время, управление) располагаются в 0-ом столбце в 8 строках по одному компоненту. Компонент продукт в 0-ой строке с расшифровкой: политический, социальный, экономический, технический, природный - измененный и интеллектуальный. Или, например: вещественный, энергетический, социальный, экономический, интеллектуальный (в зависимости от макро уровня).

4. Пространство, образуемое двумя векторами: нулевым столбцом (из категорий и общенаучных понятий) и нулевой строкой (из категорий и общенаучных понятий) и разбитое таким образом на ячейки, содержит представление обо всех процессах протекающих в ДТПС.

5. КПП на макро уровне моделирует перечень теорий (моделей), описывающих развитие и функционирование деятельно-техно-природных систем (ДТПС), включающих в себя 9 компонентов.

6. Каждая из компонентов на первом этапе представляется морфологической моделью (таблицей), построенной по определенному правилу, с программным продуктом, позволяющим производить ее обработку и интегрировать содержательное и количественное описание.

7. Полученные описания в дальнейшем используются для построения технологии принятия решений по развитию и функционированию деятельно-техно-природных систем .

8. Описание элементов технологии принятия решений на макро уровне приводится в монографии «Системные исследования мелиоративных процессов и систем. Москва, 1995» и других работах.

ОРГАНИЗАЦИЯ СИСТЕМНО – СТРУКТУРНЫХ ИССЛЕДОВАНИЙ (РЕФЕРАТИВНЫЙ ОБЗОР)

Основная особенность и характеристика системного движения (делающая его "движением", а не "направлением", "подходом" и т.п.) заключена прежде всего в том, что в нем объединяются представители самых разных профессий (инженеры, военные, педагоги, ученые, философы, математики, организаторы и управляющие), Носители разных средств и стилей мышления, разных ценностных установок и точек зрения, используя стандарты и нормы своей профессии, попрежнему стремятся к получению таких продуктов, которые были заданы как образцы в их профессии, и работают привычными для них профессиональными средствами и методами, развивают за счет системных средств и методов профессиональную организованность своей исходной мыследеятельности. Системное движение сложилось и развивается как интердисциплинарное и интерпрофессиональное образование.

Системное движение в своем становлении и развитии должно учитывать всю современную социокультурную ситуацию и исходить из предельно широкого понимания возможностей и перспектив её развития. В современной социокультурной ситуации можно выделить, по крайней мере, восемь моментов, имеющих самую непосредственную связь с системным движением.

Первый момент - это процесс все более углубляющей дифференциации наук и профессий.

Второй момент - это существование узко специализированных каналов трансляции разделенной на части предметной культуры.

Третий момент - кризис «классической» философии, вызванный осознанием того факта, что эта философия лишилась своих средств управления наукой и потеряла роль координатора в развитии наук, роль посредника, переносящего методы и средства из одних наук в другие.

Четвертый момент - оформление инженерии как особой деятельности, объединяющей конструирование с разными формами квазинаучного анализа.

Пятый (очень важный) момент - это продолжающееся выделение внутри деятельности и обособление различных производственных технологий, приобретающих самодовлеющее значение и становящихся как бы новым принципом и объективным законом в организации всей нашей жизнедеятельности и, в конечном счете, подчиняющих себе и деятельность, и природу, и поведение людей.

Шестой момент - становление, оформление и частичное обособление проектирования как деятельности особого рода. В результате еще резче встал вопрос о связи и соотношении собственно проектных и исследовательских разработок.

Седьмой момент - увеличение значения и роли во всей нашей общественной жизни организационно-управленческой деятельности. Эффективность её зависит в первую очередь от научного обеспечения.

Восьмой момент (также особенно важный) - становление и оформление наук нового типа, которые грубо можно было бы назвать "комплексными науками" Сюда нужно отнести науки, обслуживающие педагогику, проектированные, военное дело, управление и т.д. и т.п.

Строение продукта, который должен быть получен в системном движении, если его целью действительно является достижение синтеза, системные проблемы и задачи по своему происхождению и специфике являются не объективными, а предметными. Они возникают в ситуации, когда нужно соотнести и связать друг с другом разнопредметные представления одного объекта. Именно эти проблемы и задачи порождают специфическую системную технику мышления, в частности, в исследовании, проектировании, планировании и управлении. Эта техника остается действенной и эффективной только в движении от множества разрозненных односторонних представлений объекта к единому и целостному представлению. Мы должны работать с этими разными предметами, двигаясь как бы над ними и по ним, добиваясь связного описания объекта при различии и множественности фиксирующих его предметов. В этих случаях, очевидно, мы уже не можем находиться внутри этих предметов и действовать по имманентным для них законам, а должны "выскочить" за их границы, работать каким-то особым образом, связывая между собой элементы разных предметов либо для целей частной практики, либо для широких теоретических целей. Организованности мышления, которые дают нам возможность ассимилировать научные предметы и описывать объект не сквозь призму какого то одного предмета, а учитывая сразу множество предметов, особенности каждого из них, и, вместе с тем, имея особую точку зрения, отличную от каждого предмета и превращающую сами эти предметы одновременно как в функциональные элементы "машины" нашего мышления, так и в объекты нашего мышления и деятельности оперирования. Специфические организованности, решающие эти задачи, и есть организованности методологического мышления и методологической работы, которые не должны отождествляться ни с собственно философскими, ни с социально-научными формами организации мышления и деятельности.

В данном контексте методологическая работа может быть выделена и противопоставлена конкретно-научной и философской работе по шести основным признакам:

(1). Методологическая работа не есть исследование в "чистом виде"; она включает в себя также критику и схематизацию, программирование и проблематизацию, конструирование и проектирование, онтологический анализ и нормирование в качестве сознательно выделенных форм и этапов работы. Суть методологической работы не столько в познании, сколько в создании методик и проектов. Она не только отражает, но также и в большей мере создает, творит заново, в том числе - через конструкцию и проект. И этим же определяется основная функция методологии: она обслуживает весь универсум человеческой деятельности, прежде всего проектами и предписаниями. Но из этого следует также, что основные продукты методологической работы - конструкции, проекты, нормы, методические предписания и т.д. не могут проверяться и никогда не проверяются на истинность. Они проверяются лишь на реализуемость. Здесь положение такое же, как в любом виде инженерии или архитектурного проектирования. Когда мы проектируем какой-либо город, то бессмысленно спрашивать, истинен ли наш проект: ведь последний соответствует не городу, который был, а городу, который будет. Не проект, следовательно, отражает город, а город будет реализацией проекта.

Начинаем рассматривать наряду с познавательной деятельностью также инженерную, практическую и организационно - управленческую деятельность, которые ни в коем случае не могут быть сведены к получению знаний. И естественно, что методология как новая форма организации мышления и деятельности должна охватить и снять все названные типы мыследеятельности.

(2). Сделанные выше столь резкие утверждения не означают, что исследование и знание исключаются из области методологии. Наоборот, методология именно тем отличается от методики, что она до предела насыщена знаниями (в точном смысле этого слова) и включает четко ограниченное, выделенное и, можно сказать, рафинированное исследование. Методологическая работа и методологическое мышление соединяют проектирование, критику и нормирование с исследованием и познанием. При этом исследование подчинено проектированию и нормированию,

хотя может быть организовано как автономная система; но, в конечном счете, исследование в рамках методологии всегда обслуживает проектирование и нормирование, оно направляется их специфическим целям.

(3). Методология не только не отвергает научного подхода но, наоборот, продолжает и расширяет его, распространяя на такие области, где раньше он был невозможен.

Прежде всего, это проявляется в том, что методология создает очень сложные композиции из знаний разного типа, недоступные традиционной науке. В частности, она по новому сочетает и соединяет естественно - научные, конструктивно - технические, исторические и практико-методические знания. Традиционная наука избегала объединять эти четыре типа знаний, и в этом она была права, поскольку её главная задача состояла в том, чтобы создать "чистое изображение" натурального объекта. Методология создает более сложные суперструктуры, связывающие знания разных типов, и постоянно пользуется такими связями. Она создает и использует знания о знаниях, она как бы все время осознает самое себя, свои собственные структуры, - это необходимо, ибо без такого осознания формы и структуры знаний вообще и специфики разных типов знаний в частности невозможно осуществить ту связь и координацию разных типов знания, о которой было только что сказано.

(4). Вместе с тем, методология стремится соединить и соединяет знания о деятельности и мышлении со знаниями об объектах этой деятельности и мышления, или, если перевернуть это отношение - непосредственно объектные знания с рефлексивными знаниями. Поэтому объект, с которым имеет дело методология, напоминает матрешку. Фактически это особого рода связка из двух объектов, где внутри исходного для методологии объекта - деятельности и мышления - вставлен другой объект - объект этой деятельности или этого мышления. Поэтому методология всегда имеет дело с двойственным объектом - не с деятельностью как таковой и не с объектом этой деятельности как таковым, а с их "матрешечной" связкой.

Методологическое знание состоит из двух знаний; знания о деятельности и знания об объекте этой деятельности. Это просто разные знания о разном. Но суть методологического подхода как раз в том и состоит, что мы связываем и соединяем эти знания. И именно в том, как определяются и устанавливаются способы соединения этих разнотипных знаний, и заключена важнейшая особенность методологии. Ведь между деятельностью и ее объектом нет соотношения "целое-часть": деятельность не добавляется к объекту как вторая, дополняющая его часть, и, точно также, объект не является просто частью деятельности. Объект деятельности включен в деятельность многократно - и как её элемент, и как содержание других элементов, например, знаний, и как материал. Таким образом, можно сказать, что методология задает логику рефлексии, т.е. логику и правила подобного соединения разнородных знаний.

(5). Для методологии характерен учет различия и множественности разных позиций деятеля в отношении к объекту; отсюда - работа с разными представлениями об одном и том же объекте. В том числе с разными профессиональными представлениями: при этом сами знания и факт их множественности рассматриваются как объективный момент мыследеятельной ситуации.

Это крайне важное обстоятельство. Классическая философия, как и вся построенная на ней наука, исходили из представления об одном естественно истинном знании. Если одна и та же ситуация описывалась по разному в различных знаниях, то обычно ставился вопрос, какое же из них истинное. Методология, в противоположность этому, исходит из того, что одному и тому же объекту может соответствовать много разных представлений и знаний и их не имеет смысла проверять на истинность относительно друг друга, ибо они просто разные. Это важнейший принцип современного методологического мышления, который называется принципом множественности представлений и знаний, относимых к одному объекту. Но так как сам объект берется всегда предметно, т.е. всегда в связке с его представлениями, то множественность разных представлений оказывается фактом деятельной и коммутативной ситуации, объединяющей разных профессионалов. Методология начинает свою работу с представлений профессионалов об объекте и первоначально объект задан

только этим множеством представлений. Лишь затем, исходя из всей этой совокупности представлений, методолог может ставить вопрос о реконструкции объекта в том виде, как он существует "на самом деле", и производить эту реконструкцию, предлагая характеризовать объект с разных сторон, как бы в разных его проекциях. Любое антологическое представление объекта является подлинным лишь с исторически ограниченной точки зрения. И поскольку мы никогда не можем уйти от этого ограничения, то всегда должны рассматривать объект в связке с набором знаний, всегда соотносить и связывать друг с другом знания разного типа - знания об объекте и знания о знаниях. В силу этого методологического мышления пользуются всегда схемами многих знаний и в своих изображениях фиксируют множество разных знаний об одном объекте; называется это приемом многих знаний.

(6). В методологии связывание и объединение разных знаний происходит, прежде всего, не по схемам объекта деятельности, а по схемам самой деятельности. Для реконструкции объекта на основе разных представлений профессионалов у нас нет другого пути, кроме выяснения того, в чем состояла "деятельная заинтересованность" этих профессионалов. И только после того, как мы опишем мыследеятельность профессионалов, заставившую их представить объект именно так, а не иначе, и таким образом, определим те фокусы, с точки зрения которых они строили свои представления, только после этого мы можем начать собирать и сорганизовать все эти представления, но опять таки не прямо через представление о деятельности, ибо реально разные представления нужно собирать в целое и сорганизовать только тогда, когда деятельности, с которыми они связаны, входят в кооперацию друг с другом, когда они начинают с разных сторон обрабатывать объект, ставший для них единым. В этом состоит основной принцип методологического мышления: представление о сложной кооперированной деятельности выступает в качестве средства связывания разных представлений об объекте этой деятельности. И это связывание идет не столько по логике устройства и жизни рассматриваемого нами объекта, сколько по логике использования разнообразных знаний в коллективной кооперированной деятельности.

По этой причине в методологической работе бывает всегда не одно онтологическое представление, а, по меньшей мере, два: одно из них изображает структуру профессионально - кооперированной деятельности - это так называемая организационно-деятельностная онтология, а другое изображает объект этой кооперированной деятельности - это натурально - объектная онтология. Особое соединение и связь этих двух онтологических представлений составляет каждый раз специфическую особенность конкретной методологической работы.

Все названные выше моменты могут быть подытожены в одном тезисе: методологическая работа направлена не на природу как таковую, а на мыследеятельность и ее организованности, причем организованности мыследеятельности имеют как бы двойное существование: один раз в качестве элементов и компонентов мышления и деятельности, а другой раз в качестве независимых и автономных образований (как правило, искусственно - естественных), размноженных в разных формах и связываемых между собой процессами мыследеятельности. Сами "натуральные объекты" рассматриваются при этом как особые организованности, мыследеятельности, создаваемые внутри философии и естественно - научных предметов наряду с другими: естественно - научная ориентация на так называемый натуральный объект оказывается лишь одним из многих подразделений в организации наших знаний и нашего мышления.

Но наше обстоятельство - смена природной деятельности на деятельность при переходе к методологическим формам работы - ставит перед нами новый круг весьма сложных проблем: чтобы научиться работать с комплексными структурами знаний, объединяющими, с одной стороны, методические, конструктивно-технические, естественно - научные, исторические и философские знания, а, с другой стороны, знания об объектах и знания о знаниях и мыследеятельности, нужно разработать новую логику мышления, которую суммарно можно назвать логикой рефлексии. С этой точки зрения современная методология будет характеризоваться как основывающаяся на логике рефлексии.

К этому можно добавить, что сама логика рефлексии предполагает еще особые знания о рефлексии. Когда обсуждаем весь этот круг вопросов, то движемся еще в одном особом типе знаний, который может быть назван методологически рефлексивным. Многие из сделанных выше утверждений развертывались не в действительности методологии, а в действительности метаметодологии. Вместо того, чтобы осуществлять какую-либо мыслительную или деятельностную процедуру и демонстрировать её, описывали либо её саму, либо осуществляемое ею преобразование, его возможные продукты и результаты. Именно за счет этого и появлялось различие между действительностью методологии и действительностью методологической рефлексии (метаметодологии).

Специфика системного подхода может быть определена только при описании структуры и форм организации методологической работы, ибо системный подход существует только как подразделение и особая организованность методологии и методологического подхода.

Он возникает в условиях, когда приходится объединять несколько разных предметов - об этом уже говорили - и двигаться при этом в соответствии со средствами и нормами методологии. И если само выражение "системный подход" и соответствующие ему организационности мышления и деятельности появляются также и у представителей специальных наук, то это происходит за счет того, что они заимствуют средства, методы и онтологию методологической работы. Следовательно, только описывая структуру методологической работы и методологии, можем подойти к вопросу о специфике системного подхода.

Специфика системного подхода будет определяться поразному в зависимости от того, какую систему описания выберем. Если выберем описание в теории мышления, то будем определять специфику системного мышления.

Но можно описывать системный подход также и в средствах теории деятельности, тогда специфика его будет выражена и зафиксирована иначе. Таким образом, и здесь должны учитывать момент множественности возможных представлений.

После того как это зафиксировано, можно сделать следующий шаг и попробовать собрать и изобразить на схеме те особенности или принципы методологического подхода, которые формулировались ранее. Иначе говоря, теперь надо нарисовать схему системно-структурной работы, учитывая сформулированные выше принципы. На схеме блоки предметов, вырастающих над практиками разного рода, охватываются частными системно - структурными методологическими разработками и как бы надстраивается над предшествующим и ассимилирует его. Это:

- 1) слой практик (включая туда инженерно-конструкторские, организационно - управленческие, проектные педагогические и другие разработки);
- 2) слой научных, инженерных, оргуправленческих, проектных и других предметов;
- 3) слой частных методологических разработок и, наконец,
- 4) слой общей методологии.

Выше уже подчеркивали, что продуктом методологических разработок должны быть не только знания (тем более, научные), сколько методологические предписания, проекты, программы, нормы и т.п. которые будут использоваться в нижележащих слоях мышления и деятельности - в частнометодологических разработках, в предметах разного рода и в практиках. Поэтому первая и основная часть общей системно-структурной методологии должна быть не исследовательской, а конструкторской и проектной. Схематизируя этот вывод, изобразили в "теле" общей методологии над совокупностью частно-методологических разработок слой общего методологического системно-структурного конструирования и проектирования (процесс обеспечения - частнометодологических и предметных разработок общими средствами).

В специальных логико-методологических исследованиях было установлено, что во всяком научном предмете имеется, по меньшей мере, девять разных эпистемологических единиц:

- 1) проблемы,
- 2) задачи,

- 3) "опытные факты",
- 4) "экспериментальные факты",
- 5) совокупность тех общих знаний, которые строятся в этом научном предмете,
- 6) онтологические схемы и картины,
- 7) модели,
- 8) средства (языки, понятия, категории),
- 9) методы и методики.

Это набор основных блоков научного предмета.

Четыре элемента всякого научного предмета - онтологические схемы и картины, средства и методы, а также проблемы - всегда вырабатывались либо целиком за пределами научных предметов (в философии и в зародышевых структурах естественнонаучной методологии), либо же формально в рамках науки, но на деле - в захваченных ею системах философского и методологического мышления.

Но для того, чтобы блоки конструирования и проектирования, представленные в общей схеме системно - структурной методологии могли бы работать, нужно еще иметь, по крайней мере, две группы специальных знаний; во - первых, разнообразные знания (конструктивно - технические, проектно - технические, естественнонаучные и т. д.) о тех объектах, которые создаются конструктивно-методологической и проектно - методологической мыследеятельностью; это - обязательное требование всякой продуктивной работы, не имеющей прототипов: поскольку блок методологического конструирования и проектирования поставляет в научные, инженерные и управленческие предметы определенные организованности, функционирующие дальше по законам этих предметов, то для проектирования необходимо знать назначение и функции этих организованностей, требования к их морфологии и т.п.; во - вторых, методики и понятийные средства самого методологического конструирования и проектирования.

Методологические исследования - это исследования особого типа: поскольку объектами их являются не физические, химические и биологические явления, а научные предметы, т.е. знания из тех или иных наук вместе с объектами этих знаний и вместе с деятельностью порождения и использования знаний; в силу этого должны здесь говорить об исследовании, которое отличается от естественнонаучного, прежде всего объекта. Но специфика объекта изучения влечет за собой специфику средств и методов исследования, и поэтому точно так же можем и должны говорить здесь о специфике технологии методологических исследований. Напомним тезисы, выдвинутые Дж. Клиром и В.П. Садовским: « общая теория систем" является не теорией, а метатеорией; это значит, что на вопрос о том, что же является объектом понимаемой таким образом "общей теории систем", должен последовать и следует ответ: понятия, языки, методы, проблемы других наук. Главное здесь нащупано и выражено: хотя ОТС - это не естественнонаучное исследование, но все же исследование, и, будучи исследованием, оно сильно отличается от традиционного естественнонаучного исследования. Это исследование целиком входит в систему методологической работы - и этим определяется его специфика, но оно ни в коем случае не исчерпывает ни методологической работы в целом, ни даже методологического анализа, ибо наряду с ним в методологии есть и другие формы анализа. Эта форма исследования определена, во-первых, своей ориентацией на научные, инженерные, управленческие и другие предметы, а во - вторых, своей функцией обслуживания методологической конструктивной и проектной работы. Учитывая рефлексивное происхождение исследовательской работы, должны представить её в виде блока, охватывающего всё то, что исследуется.

В состав системно-структурной методологии должен войти, по крайней мере, еще один слой работы, назначение которого состоит в том, чтобы осознать и систематизировать собственную организацию методологической работы в системной области: этот блок, следовательно, организуют системно-структурную методологию как некоторое целое, связывая и объединяя всеедино методологическое системно-структурное конструирование и проектирование со всеми

обслуживающими его знаниями и методологическими системно-структурными исследованиями. Поэтому можем назвать его "методологии", или, точнее, системной авторефлексии методологии.

Этот слой работы связывает системно-структурную методологию с более широкими, объемлющими её системами-с философией диалектического материализма и всей культурой человечества, накопленной в ходе исторического развития. По сути дела это и есть слой собственно методологической рефлексии и методологического мышления, охватывающий все другие компоненты методологической работы и создающий специфику методологической организации мышления и деятельности. Дальше задача будет состоять в том, чтобы сформировать средства и методы методологического мышления как соразмерные организации его объектной области, или пространства его объектов. Если хотим рассматривать и характеризовать структуру и формы организации методологии системно-структурных исследований, то должны исходить из схемы и рассматривать методологию как сверхпредметную структуру, охватывающую как предметы, так и практики разного рода и предполагающую не одно какое-то отношение к ним, а массу разных отношений – не только исследовательское, но и конструктивное, проектное, рефлексивное, организационное и т.п.

В силу этого структурно – системная методология оказывается не просто сложной структурой и сложной системой, а гетерогенной и гетероиерархизированной системой, имеющей одновременно как ступенчато – иерархизированное, так и «матрешечное» строение.

Основную «субстанцию» (если только так можно выразиться) этой системы образует методологическая рефлексия, которая захватывает практики разного рода и обслуживающие их или независимые предметы. В этих практиках и предметах разного рода системно-структурная методологическая рефлексия выделяет системные проблемы разного рода, затем (в соответствии с разными мыслительными отношениями) – оформляется в разные виды и типы системно-структурного мышления: проектное, конструктивное, исследовательское, организационное и т. п. Все эти различные типы методологического мышления выявляются, оформляются и организуются внутри рефлексии – из её собственной субстанции и субстанции захваченных ею практик и предметов. Кроме того, все эти организованности методологического мышления сорганизуются еще друг с другом в определенные кооперативные структуры, соответствующие линиям циркуляции их продуктов в пространстве методологии. Методологическое программирование поставляет во все другие подразделения методологии программы мыслительной и практической работы, методологическое проектирование – проекты практик и предметов разного рода, методологическое конструирование – системно-структурные онтологии, средства системно-структурного анализа, т.е. системную графику, понятия, описывающие употребления этой графики в мыслительной работе, основные категории, процедуры и методы системного мышления и т.д. и т.п., а методологическое исследование – знания о системно-структурных аспектах практической и предметной работы.

Для правильного понимания всей этой организации очень важно иметь в виду, что системно-структурное методологическое исследование направлено не на системные объекты, а на системно-структурную мыследеятельность и описывает её процессы, механизмы и строение. Поэтому, кроме «системщиков», работающих в разных частных предметах и на материале практики, должны быть ещё «чистые системщики» или «системщики-методологи», которые осуществляют системно-структурное методологическое программирование, проектирование и исследование и в ходе него создают и исследуют то, что называем «структурами вообще» и «системами вообще».

В рамках системно-структурной методологии существует и должно существовать много разных типов и способов мышления и мыслительной работы, а соответственно этой – много разных позиций и, можно даже сказать, специализаций. Это будут:

- 1) организация системных практик разного рода,
- 2) разработка системных проблем в рамках частных предметов науки, инженерии, управления и т.д.,
- 3) системно-структурное программирование исследований и разработок,

4) системно-структурное проектирование,

5) системно-структурное конструирование,

6) методологическое системно-структурное исследование, описывающее системные разработки в рамках научных, инженерных и управленческих предметов и практик разного рода и, наконец,

7) методологическая авторефлексия всей области системно-структурных разработок в целом.

Это определенный проект организации методологической работы в системной области некоторые общие схемы организации методологической работы. С натуралистической точки зрения, исходящей из того, что она уже знает, что такое системный подход, а не с точки зрения методологических и деятельностных представлений, которые разворачиваются в предложении, что адекватных и эффективных системно-структурных представлений у нас сейчас нет и их ещё только надо выработать, получить, и в этом, в частности, состоит задача системного движения.

Но если эти последние утверждения правдоподобны, то могут быть только две стратегии:

1) непосредственно приступить "к делу" и начать конструировать системно-структурные представления, не зная, как это делать и что должно получиться в результате, либо же

2) спроектировать и создать такую организацию, или "машину деятельности", которая бы в процессе своего функционирования начала перерабатывать современные зародыши системно-структурных представлений в стройную и непротиворечивую систему системных взглядов и системных разработок.

Третьей стратегии не дано, хотя всегда есть путь (кстати, самый массовый и самый распространенный) нового обговаривания и переформулирования уже имеющихся представлений, созданных другими, но он не дает подлинных вкладов в культуру.

Для организации своей работы выбираем вторую. Задача - создание особой "машины мыслительности", которая будет производить системно-структурные представления: и в этом суть методологического подхода к разработке системно-структурной методологии. Если бы у нас уже были системно-структурные представления, то нам незачем было бы создавать эту "машину". В том то и дело, что у нас этих представлений ещё нет и, более того, мы даже не знаем, какими они должны быть, и чтобы как-то выйти из этой безнадежной для "натуралиста" ситуации, создаем "методологическую машину", которая будет производить нужные нам системно-структурные представления. Если это будут системно-структурные представления, то нам незачем было бы создавать эту "машину"; в том-то и дело, что у нас этих представлений ещё нет и, более того, мы даже не знаем, какими они должны быть. Чтобы как-то выйти из этой безнадежной для "натуралиста" ситуации, создаем "методологическую машину", которая будет производить нужные нам системно-структурные представления. То, что это будут системно-структурные представления, гарантируется тем, что "машина" будет ориентирована на системные проблемы и будет перерабатывать материал системной области, а то, что это будут методологические представления, гарантируется методологическим устройством самой "машины". Устройство "машины" и характер перерабатываемого ею материала, следовательно, должны гарантировать нам необходимое качество получаемых продуктов.

Если кто-то думает, что методологическое мышление, подобно научному, направляется на натуральные объекты, то, он естественно, будет считать таким материалом системно представленные натуральные объекты. Тот, кто думает, что методологическое мышление направлено на научные предметы и знания, будет считать основным материалом системного подхода системные знания и проблемы, а тот, кто считает предметом методологического анализа процедуры, методики и методы исследовательской и проектной работы, тот естественно, выделит на передний план их системные аналоги. Для нас в рамках идеи методологической организации системно-структурных исследований и разработок равно приемлемы все эти варианты: они все войдут в предлагаемую схему организации. И это, по-видимому, главное.

Важное преимущество такой организации системно-структурных исследований и разработок состоит в том, что она не отвергает ни одного из существующих вариантов предметной и

методологической работы, принимает их все и показывает место, роль и необходимость каждого. Но она, кроме того, берет их в связях и отношениях друг с другом, к их сопричастности к целому и в их зависимостях от целого и на основе этого дополнительно углубляет и развивает каждый из этих видов работы.

Кроме того, и это очень важно для понимания существа дела, эта схема устанавливает особые отношения между структурой (или устройством) "методологической машины" и захватываемым ею материалом. Характер "машины" определяется, по крайней мере, и тем, и другим, который она включает в себя, в такой же мере влияет на характер и качество её продукта, как и сама структура (или порядок и последовательность переработки материала соответствующими формами); и, более того, сам материал за счет специфического устройства этой "машины" (в особенности за счет работы блока авторефлексии) всё время оказывает давление на устройство "машины", все время перерабатывается в устройство "машины", в её формы.

И если уж останавливаться на вопросе, почему предлагаемый проект организации системно-структурной методологии и все связанные с ним представления кажутся обычно странными и вызывают много возражений, то надо указать, прежде всего, на это решение вопроса об отношении между конструкцией "машины" и захватываемым её материалом. В предлагаемом проекте системно-структурной методологии конструкция "машины" рассчитана не только на переработку захватываемого ею материала, но также на имитацию и воспроизведение морфологии этого материала (по сути дела этот принцип является дальнейшим обобщением принципа содержательности логических форм, лежащих в основании содержательно-генетической логики). Это отношение реализуется в "машине методологии" за счет методологической рефлексии и блока методологических исследований системной работы во всех видах и типах человеческой деятельности.

Вся охарактеризованная выше система методологической работы создается и организуется отнюдь не для того, чтобы решать сегодняшние проблемы, называемые "системно-структурными" (хотя по ходу дела она должна решить или чаще всего снять и эти проблемы). Система методологической работы создается для того, чтобы развивать все совокупное мышление и совокупную деятельность человечества. Непосредственным поводом для создания её служат сегодняшние проблемы, но если бы мы ограничили наши цели и задачи только ими, то это была бы во многом пустая или, во всяком случае, малоэффективная работа. Поэтому реальной целью системно-структурной методологии должно быть не устранение и преодоление той или иной группы частных проблем, а обеспечение постоянного и непрерывного системного развития деятельности. При этом, естественно, должны постоянно выявляться и фиксироваться возникающие проблемы. Но было бы ошибкой думать, что напряжения и разрывы в деятельности (или проблемы) однозначно определяют направления и способы их разрешения или, другими словами, переходы к задачам. Ничего подобного. В абстрактной возможности существует всегда бесконечное множество решений каждой проблемы, а в практическом плане - достаточно большое число существенно разных решений. Если мы объединяем проблемы и ищем одно решение для каждой из таких объединенных групп, то найти практически значимое решение конечно труднее, чем для каждой отдельной проблемы, но все равно таких решений всегда может быть несколько разных. Таким образом, напряжение, разрыв или проблема в мыследеятельности не определяют еще однозначно задачу мыследеятельности. Во многом задача определяется используемыми нами средствами; а средства есть всегда результат нашей "испорченности", нашего индивидуального вклада в историю, и именно они определяют, каким образом и за счет каких конструкций будет преодолен и снят тот или иной набор затруднений, разрывов и проблем в деятельности.

Характер системно-структурных представлений и системного подхода в целом будет определяться в первую очередь характером используемых нами средств. Предлагаем считать подлинными системно-структурными представлениями те, которые будут производиться созданной нами "машиной" системно-структурной методологии. Какой же смысл создавать новые формации

мышления и новые" машины деятельности", чтобы в результате вернуться в старые системы и к старым проблемам. Системный подход в нынешней социокультурной ситуации может быть создан и будет эффективным только в том случае, если он включен в более общую и более широкую задачу создания и разработки средств методологического мышления и методологической работы. И такой путь, как стремились показать, соответствует условиям возникновения системного подхода и традициям его развития. Поэтому важнейшей социокультурной задачей на современном этапе является соединение системного подхода с методологическим подходом и его различными вариантами, такими, как деятельностный, нормативный, типологический подходы, и обратное обогащение и развитие методологического подхода и всех его разнообразных вариантов за счет специфических средств системного подхода. И эта двусторонняя задача может быть решена с помощью и в рамках охарактеризованной выше методологической организации систем мышледеятельности.

ЛИТЕРАТУРА

1. Акофф Р.Л. О природе систем.- Изв. АН СССР. Техн. Кибернетика 1971, N 3, с. 68-75.
2. Акофф Р.Л. Планирование в больших экономических системах/Пер. С англ. под редак. И.А.Ушакова. М.:Сов. радио,1972. 367 с.
3. Блауберг И.В.,Садовский В.Н., Юдин Э.Г. Системный подход: предпосылки, проблемы,трудности.М.: Знание, 1969, с. 64.
4. Блауберг И.В.,Юдин Э.Г.Становление и сущность системного подхода.,М.: Наука,1973, 270 с.
5. Богданов А.А. Всеобщая организационная наука (тектология). 3-е изд. Москва: Берлин,1925-1929. Т. 1-3.
6. Большие системы: Теория, методология, моделирование/ Под ред. В.В.Гнеденко и др. М.: Наука, 1971,298с.
7. Волков Г. Научно-техническая революция: естествознание и обществоведение, -Правда 1978, 25 февраля.
8. Гришиани Л.М. Организация и управление. М.: Наука,1972, 432с.
9. Гуд Г.Х.,Макел Р.Э.Системотехника:Введение в проектирование больших систем. М.: Сов.радио,1962, 312 с.
10. Гуцин Ю.Ф., Щедровицкий Л.П. и др. К понятию "системное проектирование". -В кн.: Большие информационно-управляющие системы. М.:Моск. дом науч.-техн. пропаганды, 1969,82 с.
- 11.Джонсон Ф.,Каст Ф.,Розенцвейк Д. Системы и руководство: (Теория систем и руководство системами). М.: Сов. радио,1971, 47 с.
- 12.Евенко Л.И. Системный анализ-инструмент обоснования управленческих решений.-США: экономика, политика, идеология,1970, с. 51-59.
- 13.Заде Л.,Дезоер Ч. Теория линейных систем. М."Прогресс, 1970, 393 с.
- 14.Камман Р.,Фалб П.,Арбиб М. Математическая теория систем. М.: Мир, 1971, 406 с.
- 15.Квейд Анализ сложных систем.М.: Сов. радио,1969, 231 с.
- 16.Косыгин Ю.А. Методологические вопросы системных исследований в геологии. -Геотектоника, 1970,N2, с. 5-23.
- 17.Кузьмин В.П. Принцип системности в теории и методологии К.Маркса. М.:Политиздат, 1976, 261.с.
- 18.Любидев А.А.,Значение и будущее систематики.-Природа, 1971, N 2 с.112-145.
- 19.Месарович М.,Мако Д., Такахара И. Теория иерархических многоуровневых систем. М.: Мир, 1973, 342 с.
- 20.Николаев В.В. Состояние и некоторые проблемы развития системотехники . Л.:Судостроение, 1970, с. 3-38.
- 21.Общая теория систем. М.: Мир, 1966, 156 с.

-
22. Оптнер С. Системный анализ для решения деловых и промышленных проблем. М.: Сов. радио, 1969, 148 с.
 23. Поливанова С.В., Щедровицкий Г.П. Методологическая организация мышления и деятельности как условие и средство комплексной организации НИР. - В кн.: Комплексный подход к научному поиску: Проблемы и перспективы. Свердловск: УПЦ АН СССР, 1979, ч.2, с. 121-126.
 24. Проблемы исследования структуры науки. Новосибирск: НГУ, 1967, с. 238.
 25. Разработка и внедрение автоматизированных систем в проектирование (теория и методология). М.: Стройиздат, 1975, 527 с.
 26. Садовский В.Н. Некоторые принципиальные проблемы построения общей теории систем. - В кн.: Системные исследования. Ежегодник 1971 г. М.: Наука, 1972, с. 35-54.
 27. Садовский В.Н. Основания общей теории систем. М.: Наука, 1974, 279 с.
 28. Сазонов Б.В. Методологические проблемы организации проектной деятельности. - Техн. эстетика, 1973, N 4, с. 41-72.
 29. Саймон Г. Науки об искусственном. М.: Мир, 1972, 313 с.
 30. Сетров М.И. Общие принципы организации систем и их методологическое значение. Л.: Наука, 1971, 265 с.
 31. Спиркин А.Г. Фазонов В.В. Обсуждение методологических проблем исследования структур и систем. - Вопр. Философии 1964, N 1, с. 15-47.
 32. Усмов А.И. Методы построения и развития общей теории систем - В кн.: Системные исследования: Ежегодник 1973. М.: Наука, с. 146-179.
 33. Усмов А.И. Системный подход и общая теория систем. М.: Мысль, 1978, 241 с.
 34. Щедровицкий Л.П. Проблемы методологии системного исследования. М.: Знание, 1964, 56 с.
 35. Щедровицкий Г.П. и др. Проблемы системного инженерно-психологического проектирования. М.: МГУ, 1971, 124 с.
 36. Щедровицкий Г.П., Садовский В.И. К характеристике основных направлений исследования знака в логике, психологии и языкознании. Сообщ. 1-111. - В кн.: Новые исследования в педагогических науках. М.: Просвещение, 1964, вып. 2, с. 73-31, 1965, вып. 4, с. 67-72, вып. 5, с. 40-48.
 37. Щедровицкий Г.П. К характеристике более абстрактных направлений методологии структурно-системных исследований. - В кн.: Проблема исследования систем и структур. М.: АН СССР, 1965, с. 15-23.
 38. Щедровицкий Г.П. Об исходных принципах анализа проблемы обучения и развития в рамках теории деятельности - В кн.: Обучение и развитие: Материалы симпозиума. М.: Просвещение, 1966, с. 89-118.
 39. Щедровицкий Г.П., Юдин З.Г. и др. «Естественное» и «искусственное» в семантических системах. - В кн.: Семантика и восточные языки. М.: Наука, 1967, с. 48-56.
 40. Щедровицкий Г.П. Методологический смысл проблемы лингвистических универсалий. - В кн.: Языковые универсалии и лингвистическая типология. М.: Наука, 1969, с. 46-98.
 41. Щедровицкий Г.П. Проблемы построения системной теории сложного -популярного" объекта. - В кн.: Системные исследования: Ежегодник 1975. М.: Наука, 1976, с. 235-267.
 42. Щедровицкий Г.П. Системное движение и перспективы развития системно-структурной методологии. Обнинск, 1974, 56 с.
 43. Юдин Н.Г. Становление и характер системной ориентации, - В кн.: Системные исследования: Ежегодник 1971. М.: Наука, 1972, с. 18-34.

В основе концепции лежат следующие системные принципы

Система, есть то, что решает проблему.

Единственным критерием участия данного компонента в процессе может быть то, что приводит к появлению результата.

Понятие процесс - центральное понятие системных исследований.

Система определяется заданием системных объектов, свойств и связей.

Системные объекты - это вход, процесс, выход, обратная связь и ограничение.

Ограничение системы состоит из цели (функция) системы и принуждающих связей (качество функции).

Принуждающие связи должны быть совместимы с целью.

Проблемой называется ситуация, характеризующаяся различием между необходимым (желаемым) выходом и существующим выходом.

Эффективность - возможность решать проблему с помощью данной системы.

Системный подход заключается в его концептуальности, т. е. понятийном аппарате, в его идеях, подходе и установках, а не в формальном математическом аппарате.

Системный подход построен на понятиях высокого уровня общности.

Системный подход: основные правила конструктивных споров.

Процесс определяется как явление, состоящее в непрерывном изменении в течении времени.

Методология является логически и процедурно организованной последовательностью операций (воспроизводимых).

Объекты есть параметры системы.

Состояние системы описывается множеством величин по каждому системному объекту.

Свойства есть качества параметров объекта.

Качества -это внешние проявления того способа, с помощью которого получается знание об объекте или которым объект вводится в процесс.

Свойства дают возможность описывать объекты системы количественно.

Связи есть то, что соединяет объекты и свойства в системном процессе.

Термин процесс определяется как дающая данный результат общность, входящая во все объекты, свойства и связи компонентов систем.

Цель исследователя состоит в создании модели изучаемой им системы независимо от того, является ли она физической или абстрактной.

Адаптивная система - такая, в которой происходит непрерывный процесс обучения и самоорганизации.

Полная система состоит из всех объектов, свойств и связей, необходимых для достижения данной цели при данных принуждающих связях.

Формирование проблемы называется также ее определением.

Цель формирования проблемы состоит в том, чтобы установить сущность проблемы в известных терминах, а не в терминах, которые неизвестны.

Ограничение есть сумма правил, установлений и выдвинутых лично или извне руководящих принципов, определяющих границу проблемы.

Критерий является средством, с помощью которого измеряются или выбираются альтернативы.

Критерий указывает относительное достижение альтернативы в терминах других мер, таких как время, стоимость или эффективность.

Критерий есть стандарт. Риск является мерой потенциальной подверженности недостаткам.

Оптимум означает лучшее в смысле "все учтено". Он не означает "Самый лучший". Он может означать наиболее благоприятные условия, способствующие достижению цели.

Трехмерная оценка - ВРЕМЯ - СТОИМОСТЬ - ЭФФЕКТИВНОСТЬ есть предпочтение.

Термин "Выбор" означает методическую манипуляцию объектами и свойствами, позволяющую определить полный диапазон характеристик системы.

Выбор может быть определен как замена одного набора оценок, полученных из отношения набора объектов и свойств, на другой набор оценок, который должен быть оценен тем же способом.

МЕТОДОЛОГИЧЕСКИЕ ОСНОВЫ КОМПЛЕКСА МЕЛИОРАЦИЙ АГРОЛАНДШАФТА С ПОЗИЦИЙ ДЕЯТЕЛЬНО – ТЕХНО - ПРИРОДНЫХ СИСТЕМ, СООРГАНИЗУЮЩИХ СЕЛЬСКОХОЗЯЙСТВЕННУЮ И МЕЛИОРАТИВНУЮ ДЕЯТЕЛЬНОСТЬ.

Постановка темы была определена в результате анализа сложившихся пониманий в области сельского хозяйства и как многие считают, что мелиорация является подотраслью сельского хозяйства и направлена на улучшение природной среды (например, почв, приземного слоя воздуха и т.д.). Однако, наши исследования показали, что мелиорации являются более широким понятием, чем воздействие только на природную среду. Проекты, которые были реализованы в России и странах ближнего зарубежья, охватывали площади от сотен гектар до тысяч сотен гектар, а на этих площадях были расположены хозяйства, районы, области и республики. В результате этих проектов производились улучшения не только природной среды, но и социально-экономической и политической. Поэтому те оценки и те методики, при помощи которых производились оценки и принятии решений, не раскрывали существа дела. Нормативные документы в виде СНИПов не содержали в себе правильной оценки деятельности (условно) мелиораторов. Например, когда осваивались территории или массивы, площадью 100 - 600 тысяч гектар и при этом создавались условия жизни в виде жилья, детских садов, школ, больниц, библиотек, кинотеатров и театров, то оценка велась по приросту урожайности с мелиорированного (улучшенного) гектара, хотя на самом деле происходило не только улучшение сельскохозяйственных угодий, но и социальной, экономической, технической сред и интеллектуального потенциала людей, живущих на этих территориях.

С другой стороны, выше указанный анализ показал, что любая сельскохозяйственная деятельность, мелиоративная деятельность, или другие деятельности реализуются на природной среде или в природной среде в виде создания неких инженерных систем: дорог, линий электропередач, водоснабжения, канализации, сооружений промбаз и т.д., которые взаимодействуют с природной средой или, другими словами, образуют техно - природные системы.. С другой стороны, техно - природные системы не могут функционировать без участия человека, то есть, его деятельности. Поэтому, нами введено понятие деятельно - техно -природные системы (ДТПС). Представляется, что на природной среде размещается некоторое количество технических систем и при помощи человеческой деятельности реализуется целенаправленное устойчивое функционирование ДТПС.

Учитывая выше сказанное, что ДТПС могут размещаться на массивах, площадях или ландшафтах разных размеров от нескольких гектар до сотен тысяч гектар, но их компонентный состав инвариантен, то есть природная среда, технические системы и деятельность.

Обобщенный взгляд на деятельно – техно – природные системы

Иллюстрация функционирования деятельно-техно-природной системы (ДТПС) на макро уровне как взаимодействие "Общества" и " Природы" во времени выражается в виде создания "Продукта": вещественно-энергетического, социально-экономического и интеллектуального. Это множество

продуктов в свою очередь влияет на "Общество", изменяет его при взаимодействии с природой. Таким образом идет развитие общественного сознания во времени и возникает представление о совместной деятельности.

Любую систему (объект) в реальности можно представить предметом в знании о целенаправленных системах. Развитие этого предмета в мыследеятельности дает более глубокое представление о предмете на основе кибернетической структуры, нагруженной понятиями, т.е. в итоге формулируется категориально-понятийное пространство. Углубление познания при сопоставлении предмета в знаниях с объектом в реальности есть углубление знаний и расширение этого пространства в предметной области.

На основе приведенных схем для процесса выработки решений о повышении эффективности системы природопользования (СПП), если в предмете знаний выделить часть сельскохозяйственных угодий, на которых реализуется деятельность мелиораторов и сельхозпроизводителей, то через мышление и мыследеятельность по существу надо организовать деятельность этих двух производителей через цели функционирования этих объектов в реальности. При этом мышление и мыследеятельность осуществляется на основе целей и программ, которые связаны с такими показателями как духовность, проблемная ситуация, средства. В этом процессе мыследеятельность является основным звеном развития.

На основе анализа мелиоративной деятельности, которая затрагивает природную и социально-экономическую среду, предлагается ввести уровни понятий - комплекс мелиораций.

Откуда видно, что комплекс мелиораций подразделяются на мелиорации и социомелиорации, осуществляемые соответственно в природной и социально-экономической средах. Мелиорации подразделяются на сельскохозяйственные и не сельскохозяйственные мелиорации.

Сельскохозяйственные мелиорации в свою очередь подразделяются на сельскохозяйственные гидротехнические мелиорации и другие виды мелиораций. Сельскохозяйственные гидротехнические мелиорации включают в себя сельскохозяйственные оросительные и осушительные мелиорации.

Не сельскохозяйственные мелиорации подразделяются на не сельскохозяйственные гидротехнические мелиорации и другие виды не сельскохозяйственных мелиораций. Не сельскохозяйственные гидротехнические мелиорации делятся на не сельскохозяйственные оросительные и осушительные мелиорации.

В основе концепции лежат следующие системно - структурные взгляды системного подхода

Наука, изучая сущность объектов и систем с помощью категорий и понятий, дает возможность обобщить накопленные знания и объяснить исследуемые процессы и явления, причем переход от описания этих процессов и явлений к их объяснению совпадает с познанием их структур. Поэтому введение обобщающих понятий, законов, теорий на основе использования системного, в частности системно-структурного, подхода наиболее целесообразный путь к синтезу знаний.

Системно-структурный подход предполагает развитие диалектики внутреннего и внешнего, включая два аспекта: взаимодействие инженерной системы со средой и иерархичностью систем, т.е. вовлеченность любой системы как подсистемы в некоторую более сложную систему. При этом решающую роль в системно-структурном подходе играет понятие структура. Система и структура соотносительны и выступают как единство противоположностей, а сам системно-структурный подход в общем виде означает рассмотрение познавательных объектов в качестве систем элементов под углом зрения их структур.

Актуальность исследования структуры системы производства продукции на мелиорированных землях (или вообще любой системы или подсистемы) определяется тем, что в ней просматривается некоторая аналогия между важностью рассмотрения понятия "закон" и его роли в отображении

некоего явления, поскольку закон выражает существенное в явлении, а структура - существенное в системе. Структура - это такая организация объектов, которая характеризует их сущность с внутренними зависимостями.

Вместе с тем процесс системно-структурной интеграции, синтеза научных знаний также имеет ряд противоречий. Одно из них - многообразие методологических аспектов синтеза наук. Синтез межнаучных и внутринаучных исследований возможен лишь в результате раскрытия внутреннего единства всех модификаций исследований. В области технических, экономических и других наук таким звеном является кибернетическая структура процесса. Сочетание кибернетического принципа и категориально-понятийного подхода, позволит описать и объяснить процесс мелиоративной деятельности и дает возможность сделать еще один шаг на пути интеграции знаний в мелиоративной деятельности.

В основе концепции лежат следующие содержательные положения.

За последние 10-15 лет (в целом в науке и в том числе сельскохозяйственной и мелиоративной) произошли существенные продвижения в понимании современного взаимодействия человека с техно - природными системами и их взаимного влияния. Нам представляется, что современные взгляды позволяют рассматривать мелиоративную деятельность как создание деятельно - техно - природных (ДТП) систем.

Системы производства продукции на мелиорированных землях являются разновидностью подобных систем. Исследованиями за указанный период показано, что компоненты, входящие как в ДТП систему, так и в систему производства продукции на мелиорированных землях (СППнаМЗ) имеют одинаковый компонентный состав, т.е. они включают человека, инженерные системы, среду в широком смысле, материальные ресурсы и обеспечиваются управлением при надлежащем объеме информации, моделей, описывающих процессы во времени. Поэтому мы считаем, что в основании концепции А Г Р О Л А Н Д Ш А Ф Т должна лежать структура ДТП систем.

В настоящее время нет единой концептуальной основы, позволяющей согласовать программы, разрабатываемые по защите окружающей среды, программ развития агропромышленного и промышленного секторов экономики и программ социального развития общества.

Поэтому считаем целесообразным рассмотреть вопросы взаимосвязи экономики, экологии, мелиорации и информации на примере деятельностей. Концепцию развития мелиорации рассматривать как составную часть социально-экономического развития общества на этапе перехода к рыночным отношениям.

Мелиоративную деятельность необходимо контролировать, отслеживая изменения деятельно - природно-мелиоративных ситуаций, что даст возможность оперативного вмешательства для устранения отрицательных воздействий мелиоративной деятельности на мелиорированные земли и экологическую ситуацию.

Деятельно - техно - природная система (ДТПС) -это конгломерат, сочетающий в себе природную и деятельностную части, которые живут и развиваются по различным законам, и поэтому требуют специальных средств и методов для изучения. Функционирование инженерной составляющей любой ДТП системы может существенно влиять на значительную часть природного компонента и трансформировать его.

Разработка эффективных мероприятий по управлению состоянием мелиорируемых земель, прилегающих территорий и водных объектов должна базироваться на единой информационной системе сбора, обработки и хранения данных наблюдений за процессами, протекающими под влиянием водных мелиораций и включать прогнозы возможных изменений при осуществлении проектов водохозяйственной деятельности.

Выполненный анализ показывает, что и деятельностный компонент так же требует разработки своего мониторинга связанного с экономико - социально - инновационной деятельностью.

Итак, ПОЛУЧЕНИЕ НЕКОЕГО ПРОДУКТА (ИЛИ ПРОДУКТОВ) ЕСТЬ ДЕЯТЕЛЬНОСТЬ, КОТОРАЯ РЕАЛИЗУЕТСЯ ЧЕРЕЗ ДЕЯТЕЛЬНО-ТЕХНО-ПРИРОДНЫЕ (ДТП) СИСТЕМЫ.

ДТП СИСТЕМЫ МОГУТ БЫТЬ РАЗНОГО УРОВНЯ; НА УРОВНЕ: ГОСУДАРСТВА, РЕСПУБЛИКИ, КРАЯ, ОБЛАСТИ, РАЙОНА, ХОЗЯЙСТВА (СОВХОЗА, КОЛХОЗА, ФЕРМЫ, ИНДИВИДУМА).

ПОЭТОМУ ДТП СИСТЕМЫ ОБРАЗУЕТ РЯД ОТ БОЛЬШИХ И СЛОЖНЫХ ДО ПРОСТЫХ (ЧЕЛОВЕК, ЛОПАТА, ЗЕМЛЯ И Т.Д.).

ВЫПОЛНЕННЫЙ АНАЛИЗ ДТП СИСТЕМ РАЗЛИЧНЫХ УРОВНЕЙ ПОКАЗАЛ, ЧТО ИХ КОМПОНЕНТНЫЙ СОСТАВ ИНВАРИАНТЕН, Т.Е. ОСТАЕТСЯ ПОСТОЯННЫМ И ВКЛЮЧАЕТ ТРИ ОСНОВНЫХ БЛОКА:

СРЕДУ (В ШИРОКОМ СМЫСЛЕ: ПОЛИТИЧЕСКУЮ, СОЦИАЛЬНУЮ, ЭКОЛОГИЧЕСКУЮ, ТЕХНОЛОГИЧЕСКУЮ И ПРИРОДНУЮ КАК ОКРУЖАЮЩУЮ);

КОМПЛЕКС ИНЖЕНЕРНЫХ СИСТЕМ И ПРИРОДНУЮ СРЕДУ КАК ОБЪЕКТ ВОЗДЕЙСТВИЯ и взаимодействия.

ПРИРОДНАЯ СРЕДА, НА КОТОРОЙ РАЗВОРАЧИВАЮТСЯ ВСЕ ДЕЙСТВИЯ, КОМПЛЕКС ИНЖЕНЕРНЫХ СИСТЕМ, ПРИ ПОМОЩИ КОТОРЫХ РЕАЛИЗУЮТСЯ ТЕХНОЛОГИИ ПОЛУЧЕНИЯ НЕКИХ ПРОДУКТОВ И САМ ЧЕЛОВЕК И ЕГО ДЕЯТЕЛЬНОСТЬ.

КАЖДАЯ ИЗ ЭТИХ КОМПОНЕНТ РАСКРЫВАЕТСЯ НА СЛЕДУЮЩЕМ УРОВНЕ СВОИМ КОМПОНЕНТНЫМ СОСТАВОМ.

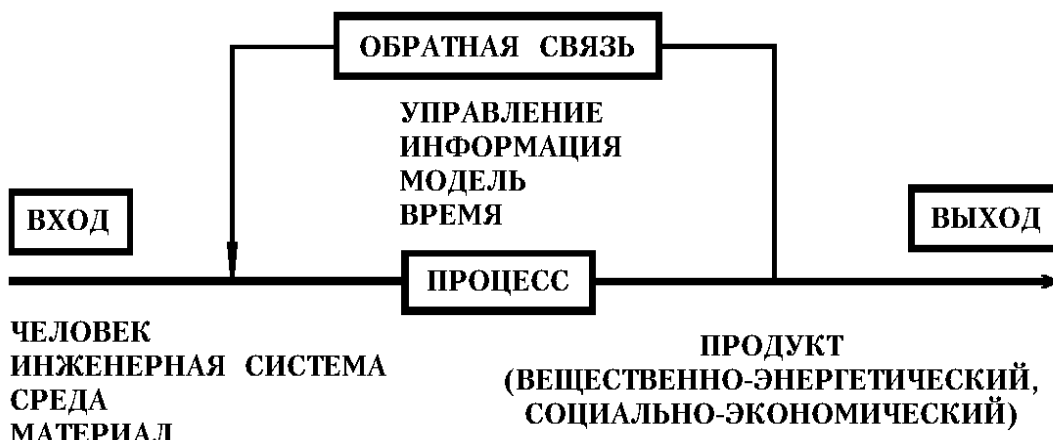
ТАК ПРИРОДНАЯ СРЕДА ОПИСЫВАЕТСЯ КЛИМАТИЧЕСКИМИ, ПОЧВЕННЫМИ, ГИДРОГЕОЛОГИЧЕСКИМИ, ГЕОЛОГИЧЕСКИМИ, ИНЖЕНЕРНО-ГЕОЛОГИЧЕСКИМИ, ГИДРОЛОГИЧЕСКИМИ УСЛОВИЯМИ. КАЖДОЕ УСЛОВИЕ РАСКРЫВАЕТСЯ ЧЕРЕЗ СВОИ ПОКАЗАТЕЛИ, НАПРИМЕР, КЛИМАТИЧЕСКИЕ: ЧЕРЕЗ ТЕМПЕРАТУРУ, ОСАДКИ И Т.Д.

ИНЖЕНЕРНЫЕ КОМПЛЕКСЫ ОБРАЗУЮТСЯ В ЗАВИСИМОСТИ ОТ УРОВНЯ РАССМОТРЕНИЯ В СЕТИ ЖЕЛЕЗНЫХ ДОРОГ, ЭНЕРГОСНАБЖЕНИЯ, ВОДОСНАБЖЕНИЯ, ПРОМСТРОЙБАЗЫ И Т. Д. ДО ЛОПАТЫ В РУКАХ ИНДИВИДУМА.

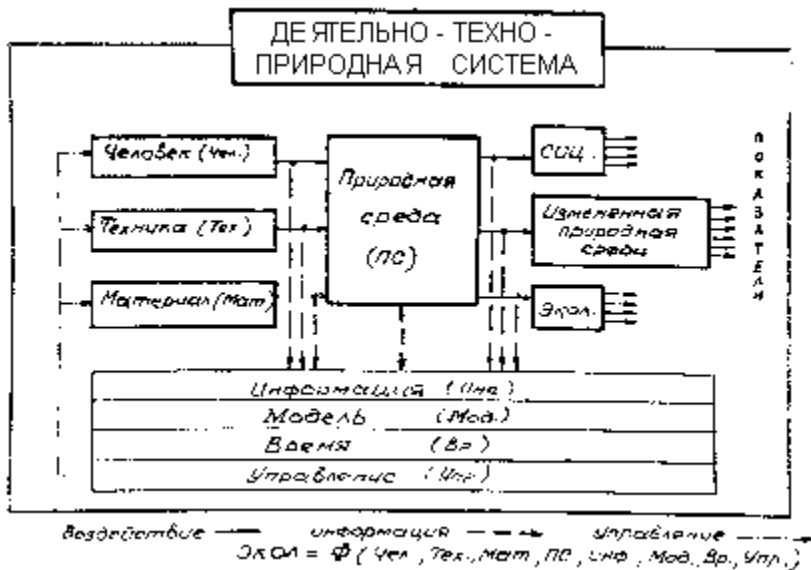
ДЕЯТЕЛЬНОСТЬ – ЭТО, ВО-ПЕРВЫХ, САМ ЧЕЛОВЕК (С ЕГО НАВЫКАМИ, УМЕНИЯМИ, ЗНАНИЯМИ И Т.Д.), ВО-ВТОРЫХ, ЭТО СПОСОБЫ УПРАВЛЕНИЯ ВО ВРЕМЕНИ, ИНФОРМАЦИЯ, МОДЕЛИ, КОТОРЫМИ ОН МЫСЛИТ И ИСПОЛЬЗУЕТ В СВОЕЙ ДЕЯТЕЛЬНОСТИ И, НАКОНЕЦ, ПРОДУКТ ДЕЯТЕЛЬНОСТИ (ИНТЕЛЛЕКТУАЛЬНЫЙ, ВЕЩЕСТВЕННЫЙ, ЭНЕРГЕТИЧЕСКИЙ, СОЦИАЛЬНЫЙ, ЭКОНОМИЧЕСКИЙ И Т.Д.).

СТРУКТУРУ ДЕЯТЕЛЬНОСТИ МОЖНО ПРЕДСТАВИТЬ В САМОМ УПРОЩЕННОМ ВИДЕ ВИНАРОВСКОЙ СХЕМОЙ:

ВХОД, ПРОЦЕСС, ВЫХОД, ОБРАТНАЯ СВЯЗЬ.



Морфологическую структуру деятельно-техно-природных систем, учитывая изложенное выше, представим в виде



ЕСЛИ ВВЕДЕННЫЕ КОМПОНЕНТЫ РАЗНЕСТИ СЛЕДУЮЩИМ ОБРАЗОМ:

НА ВХОДЕ ПОДАЕТСЯ - ЧЕЛОВЕК, ИНЖЕНЕРНАЯ СИСТЕМА, МАТЕРИАЛЬНЫЙ РЕСУРС, СРЕДА;
НА ВЫХОДЕ - ПРОДУКТ (В ШИРОКОМ СМЫСЛЕ); ОБРАТНАЯ СВЯЗЬ СОДЕРЖИТ: ИНФОРМАЦИЮ,
МОДЕЛЬ, ВРЕМЯ, УПРАВЛЕНИЕ, ТО МОЖНО ПОСТРОИТЬ КАТЕГОРИЙНО --ПОНЯТИЙНОЕ
ПРОСТРАНСТВО, РАСПОЛОЖИВ: ВХОД И ОБРАТНУЮ СВЯЗЬ ПО ВЕРТИКАЛИ, А ПРОДУКТ
ДЕЯТЕЛЬНОСТИ ПО ГОРИЗОНТАЛИ, получим категорию-понятийную матрицу ДТПС.

Т.е. продукт (в широком смысле) является функцией восьми компонент: $P = \Phi(\text{Чел.}, \text{Среда}, \text{Инж. сист.}, \text{Матер.}, \text{Инф.}, \text{Модель}, \text{Время}, \text{Упр.})$.

Каждая клеточка категорию - понятийной матрицы является неким элементарным процессом. Например: 1.1 - влияние человека на получение компонента измененной природной среды и т.д. При этом каждую компоненту можно рассматривать с двух позиций: с физической (материальной) и абстрактной, например, человек с его физическими характеристиками или его формализованное представление. Таким образом, мы имеем два категорию - понятийных дерева: дерево действия и дерево продукта, которые могут иметь понятия следующих уровней, например: среда: политическая, социальная, экономическая, техническая, природная, а, например, природная среда: климатические, гидрологические, почвенные, гидрогеологические, инженерно - геологические, биологические условия. В свою очередь, например, климатические условия описываются: радиацией, осадками и т.д.

Т.е. продукт (в широком смысле) является функцией восьми компонент: $P = \Phi(\text{Чел.}, \text{Среда}, \text{Инж. сист.}, \text{Матер.}, \text{Инф.}, \text{Модель}, \text{Время}, \text{Упр.})$.

Каждая клеточка категорию - понятийной матрицы является неким элементарным процессом. Например: 1.1 - влияние человека на получение компонента измененной природной среды и т.д. При этом каждую компоненту можно рассматривать с двух позиций: с физической (материальной) и абстрактной, например, человек с его физическими характеристиками или его формализованное представление. Таким образом, мы имеем два категорию - понятийных дерева: дерево действия и дерево продукта, которые могут иметь понятия следующих уровней, например: среда: политическая, социальная, экономическая, техническая, природная, а, например, природная среда: климатические, гидрологические, почвенные, гидрогеологические, инженерно - геологические, биологические условия. В свою очередь, например, климатические условия описываются: радиацией, осадками и т.д.

Категориально-логическая матрица
ДЕЯТЕЛЬНО - ТЕХНО - ПРИРОДНОЙ СИСТЕМЫ

Результат Деяствие	ПРОДУКТ (П)											
	Изменная природная среда (ИПС)				Экономическая среда (ЭС)				Социально-экономическая среда (СЭС)			
Человек (Чел.)												
Техника (Тех.)												
Материал (Мат.)												
Среда (Ср.)												
Информация (Инф.)												
Модель (Мод.)												
Время (Вр.)												
Управление (Упр.)												

$P = \Phi(\text{Чел., Тех., Мат., Ср., Инф., Мод., Вр., Упр.})$

Разработка и внедрение экспертных систем в области мелиораций.

В настоящее время наиболее трудоемким процессом в производстве и управлении является процесс оценки ситуации и принятия решения. Это объясняется в первую очередь постоянно возрастающим объемом информации, которую необходимо учитывать для повышения объективности оценки ситуации. С другой стороны, знания, позволяющие эксперту получать качественные и эффективные решения поставленных задач, являются в основном эвристическими, экспериментальными, неопределенными, имеющими некоторую степень правдоподобия. Это, во-первых, ставит на первый план необходимость комплексной автоматизации процесса сбора информации и принятия решения, во-вторых, обуславливает высокую сложность создания соответствующих автоматизированных систем.

Действующие ныне принципы организации обработки информации были заложены более 30 лет назад. За это время потоки и объемы информации возросли более чем в 5 раз. При таких темпах накопления информации прежние способы ее обработки не могут удовлетворить потребностей в доступе к новым знаниям. В настоящее время уровень вовлечения новых знаний в оборот научно-производственной деятельности составляет около 2%. Бурный процесс совершенствования программно-технического обеспечения автоматических технологий обработки информации привел к появлению экспертных систем как интеллектуальных систем обработки информации на базе современных высокопроизводительных ЭВМ, разрабатываемых с использованием методов искусственного интеллекта. Связывая мощные компьютеры с богатством человеческого опыта, экспертные системы повышают ценность экспертных знаний, делая их широко применимыми, стимулирует повсеместные усилия по сбору, упорядочению, обмену и использованию прикладных человеческих знаний.

Основными преимуществами экспертных систем над человеком являются их постоянство, непротиворечивость знаний, они легко передаются, документируются и уточняются. Экспертные системы также как человек ошибаются и обладают способностью учиться на своих ошибках.

Применение передового опыта, накопленного в рассматриваемой области знаний, к решению поставленной задачи, опыта наиболее квалифицированных экспертов. Возможность объективной оценки состояния знаний в данной предметной области и на основе этого определение наиболее перспективных направлений развития этих знаний. Обучение и тренировка специалистов. Это составляет основной, но далеко неполный перечень свойств и возможностей, позволивших экспертным системам выйти на первое место среди различных методов построения автоматизированных технологий обработки информации.

Для обработки научно-технической информации такое развитие технических средств и технологий обработки информации означает необходимость разработки новых принципов и способов представления содержательной, смысловой информации научно-технического документа, а не только его библиографических реквизитов. Отображением смысловой информации, содержащейся в документе, в настоящее время считается составляемый автором реферат. Такое представление является субъективным и не всегда отражает реальную ценность работы. Автореферирование должно осуществляться на общих принципах представления знаний, принятых в данной предметной области.

Разработанная система «Карта знаний науки» создается как интегрированная база знаний, дающая общий методологический подход к обработке и представлению знаний в

определенной сфере науки и производства. Предлагаемый метод представления знаний дает возможность организовать быстрый поиск научно-технической информации за счет последовательной локализации области знаний. Эксперименты, проведенные в Московском государственном университете природообустройства с группами студентов и аспирантов, показали, что поиск литературы по заданной тематике ускоряется в 200 - 300 раз по сравнению с традиционными. Автоматизация интеллектуального труда человека за счет внедрения данного метода на современных ЭВМ еще больше ускорит поиск и подбор литературы по заданной теме, предоставив возможность человеку больше времени уделить непосредственно творческому процессу.

Наибольший эффект применение экспертных систем должно принести в таких областях человеческой деятельности как мелиорация. Это обусловлено тем, что мелиорация как область знаний является наименее формализованной. Подавляющее большинство знаний здесь являются эмпирическими и трудно поддаются формализации. Стоящие перед мелиоративной наукой задачи являются трудными или вообще недоступными для традиционного программирования. Вследствие чего в настоящее время применения ЭВМ в данной области находится на очень низком уровне в сравнении с другими областями человеческой деятельности. Программы, разработанные для ЭВМ, имеют как правило локальное значение, используются в конкретных задачах конкретного региона и не тиражируются, что резко снижает экономический эффект при их внедрении.

Экспертные системы в мелиорации позволяют эффективно решать такие задачи как экспертиза проектов мелиоративных систем, экологическая экспертиза, оптимальное использование водных ресурсов в условиях их ограниченности, оценка и прогноз состояния земель в отдельных регионах с учетом хозяйственной деятельности человека и т.д.

Создание современных экспертных систем в ключевых областях мелиоративной науки наряду с широким распространением интегрированных банков данных об окружающей среде позволит поднять уровень мелиоративной науки на качественно новую ступень развития.

1. Экспертные системы как эффективное средство решения не формализуемых и трудно формализуемых задач.

1.1 Традиционные средства программирования и не формализуемые и трудно формализуемые задачи.

До сих пор перенесение умения специалиста-человека в машинную программу было утомительной и долгой процедурой. Это связано с тем, что при традиционном методе программирования проблема представлена через описание метода ее решения непосредственно процедурно ориентированным языком. Другими словами алгоритмом решения служит описание проблемы и решение всецело зависит от самой проблемы. При этом, решая различные проблемы, человеку необходимо каждый раз вводить в ЭВМ одни и те же знания, представление которых изменяется с целью адаптации к алгоритму решения конкретной проблемы. В этом случае написание программы в большинстве своем сводится к искусству нахождения наиболее подходящей формы представления неоднократно введенных ранее в ЭВМ знаний для решения конкретной проблемы. Такой подход все в большей степени порождает уже достаточно ярко проявившиеся трудности в развитии процесса компьютеризации человеческой деятельности. Это, во-первых, нерациональное использование ресурсов ЭВМ за счет многократного представления в памяти одних и тех же знаний. Во-вторых, резко затрудняет накопление, классификацию и дальнейшее использование введенных знаний из-за многообразия форм представления этих знаний. В-

третьих, нерациональное использование человеческого интеллекта, до сих пор требующего от общества создания и поддержания на высоком профессиональном уровне многочисленной и постоянно увеличивающейся группы людей - разработчиков программного обеспечения для ЭВМ.

Наряду с трудностями, связанными с общей методологией использования знаний при решении задач на ЭВМ, существуют сложности с представлением в ЭВМ конкретных знаний. Это связано в первую очередь с точностью представления фактов и достоверностью или правдоподобностью знаний. В частности, мелиоративная обстановка описывается многообразием различных по типу параметров. Здесь присутствуют и достаточно точные и однозначные сведения о географическом положении изучаемого района, его названии и т.п. Присутствуют так же данные влажность и засоленность почвы, уровень грунтовых вод и другие параметры, полученные в результате полевых измерений и содержащие в себе ошибки и неточности, заложенные в методах и технологии получения этих данных. Наряду с неточными, но достоверными данными, приходится иметь дело с величинами, имеющими определенную степень достоверности. Так, например, все прогнозные метеорологические данные, усредненные показатели ландшафта, почвенных характеристик, водного режима являют собой пример ненадежной недостоверной информации. Эту информацию необходимо использовать с известными допущениями. Результаты, основанные на этой информации, обладают определенной достоверностью, которая понятна человеку, но трудно объяснима машине в терминах традиционного программирования. В отличие от традиционных методов программирования, при использовании которых учет различного рода неточностей и неопределенностей резко усложняет процесс разработки программ для ЭВМ, а подчас делает его невозможным, использование методов искусственного интеллекта при разработке экспертных систем является единственным известным сегодня путем развития средств автоматизации научных исследований и проектирования на базе современных ЭВМ. Наряду с использованием фактов, точность и достоверность которых является предметом отдельного изучения, мелиорация как наука часто оперирует методами и законами, степень достоверности применения которых имеет определенную величину. Так, например, только для расчета параметров дренажа широко применяются, по крайней мере три схемы. Это схемы Костякова А.Н., Аверьянова С.Ф., Шестакова В.М., целесообразность применения которых в ряде случаев может быть обоснована только с использованием метода экспертных оценок. Традиционные методы программирования предоставляют два способа решения такой проблемы. Выбор одной схемы для всех случаев расчета резко снижает достоверность расчета с невозможностью оценки степени этого снижения. Ввод в определенной форме решения эксперта об использовании той или иной схемы в каждом конкретном варианте расчета отрицательно влияет на степень автоматизации расчетов.

1.2 Структура и состав экспертной системы.

В основе экспертной системы находится обширный запас знаний о конкретной проблемной области, организованной в большинстве случаев как некоторая совокупность правил, которые позволяют сделать заключения на основе исходных данных или предположений. Такой подход представляет собой революционные изменения, в корне меняющие традиционную схему

ДАнные + АЛГОРИТМ = ПРОГРАММА

на новую

ЗНАНИЯ + ЛОГИЧЕСКИЙ ВЫВОД = СИСТЕМА.

В настоящее время экспертные системы получили достаточно широкое распространение в различных областях человеческих знаний, что свидетельствует об

универсальности методов, лежащих в основе разработки экспертных систем. Особенностью этих методов является то, что они позволяют разрабатывать инструмент, облегчающий мышледеятельности человека, автоматизирующий в первую очередь наиболее рутинные процессы такой деятельности и тем самым освобождающий необходимое время человека для действительно творческой работы.

Для выполнения своих функций современная экспертная система должна включать в себя следующие компоненты:

1. Базу знаний - организованную определенным образом область памяти ЭВМ, содержащую знания как совокупность отношений между данными (фактами) из заданной предметной области. Эти отношения чаще всего представлены как правила вывода, представляющие собой долговременную информацию о том, как выводить новые факты или гипотезы из того, что сейчас известно.
2. Рабочую память (базу данных) - область памяти ЭВМ, в которой хранятся различного рода данные, факты, объекты и т.д., заданные человеком или выведенные при помощи правил, хранящихся в базе знаний. Это наиболее подверженная изменениям область памяти ЭВМ, отражающая современные представления в обслуживаемой экспертной системой области знаний (предметной области).
3. Интерпретатор или машину вывода - комплекс программ, решающий на основе имеющихся в системе знаний и данных, предъявляемую задачу. Решение конкретной задачи включает в себя два органически связанных и дополняющих процесса пополнение недостающих в рабочей области фактов, используя диалог с человеком, и получение новых фактов с использованием логики, порожденной системой правил заложенных в базе знаний правил вывода. В реальном мире экспертной системы как правило приходится работать в условиях той или иной степени неопределенности и неуверенности в полной достоверности фактов и данных. Такая ситуация требует использования методов нечеткой логики, использующих коэффициенты уверенности, многозначность, вероятностные оценки событий и т.п. Успех применения подобного рода систем обусловлен, по-видимому, тем, что организация знаний играет более важную роль, чем связанные с ними числовые значения.
4. Лингвистический процессор, осуществляющий диалоговое взаимодействие с пользователем на языке близком к естественному языку. Лингвистический процессор производит синтаксический и семантический анализ полученных от пользователя информации и синтез сообщений пользователю. При этом процесс взаимодействия пользователя с экспертной системой не сводится к обмену изолированными парами предложений типа «запрос-ответ», а представляет собой разветвленный диалог, в котором инициатива постоянно переходит от одного участника к другому. Это обусловлено тем, что для ответа на поставленный вопрос, как пользователь, так и экспертная система чаще всего нуждаются в дополнительной информации. Запрос этой информации пользователем у экспертной системы и наоборот приводит к резкому усложнению диалога, что делает практически невозможным разрабатывать заранее так называемый сценарий диалога, что повсеместно практиковалось и практикуется сейчас при создании автоматизированных систем управления традиционного типа.
5. Комплекс программ, осуществляющий процесс приобретения знаний. Основной особенностью современных экспертных систем является то, что данная компонента реализует возможность автономного без специального участия человека приобретения знаний в процессе решения задач и ввода новых данных. Для этого в современных развитых системах существует возможность создания базы метазнаний. Данная база содержит правила, алгоритмы и процедуры, описывающие процесс приобретения знаний. Одним словом, база метазнаний содержит знания о знаниях, знания о том, как приобретать новые знания.
6. Объяснительную компоненту, предназначенную для того, чтобы в случае необходимости по запросу от пользователя объяснить действия системы и ответить на вопросы о том, почему некоторые заключения были сделаны или отвергнуты. В системе, основанной на использовании правил, ответ обычно получается путем прослеживания еще раз тех шагов рассуждения, которые привели к

данному вопросу или данному заключению. Важность данной компоненты заключается в первую очередь в повышении доверия пользователя к системе.

Практика разработки экспертных систем показала, что это тот необходимый набор компонент, который позволяет достаточно эффективно моделировать мыследеятельности при принятии решений в соответствующей предметной области. Причем мощность экспертной системы обусловлена в первую очередь мощностью базы знаний и возможностью ее пополнения.

1.3 Разработка и внедрение экспертных систем.

Приступая к разработке экспертных систем (ЭС) в области мелиорации, необходимо убедиться, во-первых, в возможности построения таких систем и, во-вторых, в оправдании применения таких систем.

Наличие большого количества экспертов в области мелиорации - людей, имеющих, по общему мнению, огромный опыт работы в этой области, дает большой шанс в пользу возможности разработки ЭС в этой области. В отличие от предыдущих этапов развития мелиорации, когда основными вопросами построения мелиоративных систем были вопросы физической реализации тех или иных проектов, в настоящее время эти проблемы в основном получили свое принципиальное решение. Сейчас мелиорация как отрасль народного хозяйства имеет современную материальную базу, оснащена не уступающей мировым стандартам высокоэффективной техникой, укомплектована образованными кадрами, имеющими огромный опыт успешной реализации различных проектов. Таким образом, проблемы мелиорации в настоящее время связаны в основном с эффективностью использования накопленного потенциала, с комплексностью постановки и решения проблем, обусловленной требованием перехода от создания мелиоративных систем к созданию систем сельскохозяйственного производства на мелиорируемых землях. Но основным требованием, предъявляемым современным обществом к мелиорации, является проблема экологической чистоты проектов, перехода от систем повышения интенсификации использования человеком природных ресурсов к системам гармонизации взаимодействия с природой. Решение подобных проблем требует высокоинтеллектуальной работы при анализе ситуации и принятии решения, что также говорит за возможность и необходимость создания ЭС в области мелиорации. Структура и состав мелиоративной науки, объединяющей в себе большое количество разделов, достаточно независимых и развитых до уровня самостоятельных наук, позволяет надеяться на то, что, в случае появления задач достаточно сложных и трудных для инженерии знаний, их всегда можно будет разбить на более мелкие задачи. Такой подход позволит иметь как бы несколько уровней интеллектуальной системы, где более высокий уровень координирует подчиненные задачи и питает свою базу знаний от них.

2. *Принципы построения базы знаний в системе производства продукции на мелиорируемых землях.*

Основные принципы организации знаний применительно к системе производства продукции на мелиорируемых землях (СПП на МЗ) были разработаны в 1984 году. В работе, используя метод системного подхода, было проведено детальное исследование мелиоративных процессов и систем. В частности, проведена разработка категориально - понятийной структуры СПП на МЗ. Были выделены следующие основные категории и общенаучные понятия: ЧЕЛОВЕК, ИНЖЕНЕРНАЯ СИСТЕМА, МАТЕРИАЛ, СРЕДА, УПРАВЛЕНИЕ, ИНФОРМАЦИЯ, МОДЕЛЬ, ВРЕМЯ, ПРОДУКТ. В дальнейшем под различными системами,

входящими в понятие **ИНЖЕНЕРНАЯ СИСТЕМА** мы будем понимать собственно инженерные системы как системы машин, механизмов и других материальных объектов, созданных человеком для производства определенного продукта. Наиболее существенным с точки зрения построения базы знаний является положение о том, что: «Важным условием категориального синтеза теории является последующее выражение категорий через частные понятия. Для этого, наряду с кибернетической категориально-понятийной структурой процесса производства сельскохозяйственной продукции на мелиорируемых землях, нам представилось целесообразным отобразить эту категориально - понятийную структуру в виде категориально - понятийной матрицы, что является одним из возможных подходов построения основ обобщающей теории мелиоративной деятельности или мелиораций.» Так, например, понятие **МЕЛИОРАТИВНАЯ СИСТЕМА**, в свою очередь, входящая в категорию **ИНЖЕНЕРНАЯ СИСТЕМА**, может быть выражена через понятия инженерная СИСТЕМА **СЕЛЬСКОХОЗЯЙСТВЕННОГО ПРОИЗВОДСТВА** и инженерная **ГИДРОМЕЛИОРАТИВНАЯ СИСТЕМА**. Последняя таким же образом включает понятия инженерная **ОРОСИТЕЛЬНАЯ СИСТЕМА** и инженерная **ОСУШИТЕЛЬНАЯ СИСТЕМА** и т.д. Такой процесс представления понятия через совокупность понятий, входящих в него, будем называть детализацией понятия. Категориально-понятийной матрицей (КПМ) нулевого уровня или основной КПМ будем называть матрицу, каждой строке которой соответствует одно из понятий: **ЧЕЛОВЕК**, **ИНЖЕНЕРНАЯ СИСТЕМА**, **МАТЕРИАЛ**, **СРЕДА**, **УПРАВЛЕНИЕ**, **ИНФОРМАЦИЯ**, **МОДЕЛЬ**, **ВРЕМЯ**, влияющие на процесс производства продукции. Эти понятия и их детализации различной степени в дальнейшем будем называть исходными понятиями. Единственному столбцу КПМ нулевого уровня будет соответствовать понятие **ПРОДУКТ** как результат процесса производства. Это понятие и его детализации различной степени в дальнейшем будем называть понятием продукта.

Графом категориально-понятийных матриц или просто графом матриц будем называть ориентированный граф, корневой вершиной которого является основная КПМ. Дуга графа матриц будет означать шаг детализации одного из понятий, входящих в КПМ исходной вершины, и направлена в вершину, соответствующую КПМ, полученной в результате детализации.

В КПМ на пересечении строк и столбцов фиксируется определенная модель зависимости понятия продукта от исходного понятия. Таким образом, в процессе детализации понятий и построении графа матриц происходит детализация моделей зависимости или конкретизация знаний. Доходя до определенного уровня конкретизации, знания как отношения между понятиями, входящими в данную теорию, приобретают вид достаточно простой для представления и использования в машинной технологии обработки знаний.

Видимо процесс детализации понятий можно прервать на уровне, где отношения между понятиями приобретают либо вид некоторого алгоритма вычисления параметров понятия продукта в зависимости от значений параметров исходных понятий, либо правил нахождения этих параметров вида **ЕСЛИ ... ТО ...**

Для представления знаний в других областях человеческой деятельности более целесообразными могут быть другие виды машинного представления знаний, основанные, например, на фреймах или исчислении предикатов. В любом случае ясно, что выбор вида представления знаний необходимо произвести заранее для уточнения критерия определения уровня детализации. Однако, на любом уровне детализации возможно представление знаний в содержательной форме, в виде организованных по определенным правилам текстов, описывающих отношения между понятиями данного уровня. Процесс детализации может не доходить до элементарных уровней, указанных выше, а обрываться раньше в силу объективных причин, обусловленных уровнем развития данной области науки.

В качестве основных преимуществ вышеописанного принципа построения базы знаний можно отметить такие как наглядность, простота выделения подразделов теории как подграфа графа категориально-понятийных матриц и достаточно высокий уровень разработки методов обработки подобных структур на ЭВМ в настоящее время.

3. Построение категориально-понятийного пространства.

Представление знаний с использованием аппарата категориально - понятийных матриц основано на построении двух деревьев - дерева ПРОДУКТА (П) и дерева ДЕЙСТВИЯ (Д). Процесс построения этих деревьев заключается в том, что каждая вершина соответствующего дерева обозначает некоторое понятие из соответствующей области (ДЕЙСТВИЯ или ПРОДУКТА) системы категориально - понятийных матриц. При этом подчиненная вершина соответствует понятию, детализирующему понятие, соответствующее вершине более высокого уровня. Другими словами понятие, соответствующее родительской вершине, обобщает понятия, соответствующие подчиненным вершинам.

Далее рассмотрим декартово произведение деревьев П (ПРОДУКТА) и Д (ДЕЙСТВИЯ) как множество пар вида (п,д), где п - вершина дерева П, а д - вершина дерева Д. Это произведение назовем категориально-понятийным пространством (КПП). В этом случае всякому отношению между вершинами деревьев П и Д будет соответствовать некоторое подмножество категориально-понятийного пространства.

Для построения представления знаний в области мелиорации засоленных земель целесообразно рассмотреть подпространство КПП, включающее в себя понятия из П и Д, входящие в указанную предметную область. При этом, подпространством КПП будем называть декартово произведение двух поддеревьев соответственно из П и Д, а суммой подпространств их простое объединение как подмножеств. Заметим очевидные свойства подпространств. Два подпространства либо не пересекаются, либо одно полностью содержится в другом. Это свойство непосредственно вытекает из аналогичного свойства поддеревьев. Другое важное свойство состоит в том, что подпространства не пересекаются если, корневые вершины поддеревьев, являющихся проекциями этих подпространств на П или Д, хотя бы в одном из этих деревьев лежат на одном уровне. Сумма таких подпространств и будет в дальнейшем рассматриваться как предметная область и, в частности, система знаний в области мелиорации засоленных земель.

Каждой точке (п,д) из КПП поставим в соответствие совокупность знаний о том как, в результате каких действий, входящих в понятие «д», получается продукт «п» и назовем его элементом базы знаний $Z(п,д)$. Заметим, что $Z(п,д)$ может быть пуст, что отражает естественный факт отсутствия какого-либо знания. На самом деле содержимое элемента КПП суть совокупность отношений между атрибутами понятий, входящих в «д» и «п», записанных на определенном языке (языках).

3.1. Структура элемента базы знаний.

Элемент базы знаний содержит два типа знаний. Первый тип знаний - декларативный, представляет собой краткое содержание научных трудов, основные идеи и выводы, отражающие современный уровень знаний в данной области. К этому типу также могут относиться различные описания на содержательном уровне известных науке методов и алгоритмов, относящихся к данной теме. Другой тип - это знания продукционного типа, позволяющие на основе существующих фактов получить новое знание или проверить выдвигаемые гипотезы. Этот тип может состоять из набора правил и процедур, устанавливающих зависимости между параметрами продукта и действия.

Научные труды представляются в базе знаний своими рефератами, составленными по определенным правилам в соответствии с формулой исследования, которая включает в себя следующие правила:

- а) Дать четкое определение метатеории, которая является исходной по отношению к теории, развиваемой в исследовании: после слов «дополняющее известные знания», завершающих описание метатеории указать, чем существующая теория не отвечает требованиям практики, т.е. охарактеризовать цель исследования (неизвестное).
- б) Описать, какие методы использовались для познания неизвестного.
- в) Описать развиваемую теорию в конечном состоянии, т.е. в том виде, который она приобрела после проведения исследования.
- г) Определить практическую значимость результатов, полученных за счет применения усовершенствованной теории.
- д) Установить область возможного применения получившей развитие теории и получаемого при этом положительного эффекта.

Все научные труды, содержащиеся в базе знаний, должны сопровождаться следующими признаками.

Определяемые автором: наличие эксперимента, использование известной методики эксперимента, использование новой (автора) методики эксперимента, подтверждение гипотезы экспериментом или нет, наличие теоретической методики расчета (авторской, известной), наличие эмпирической методики расчета (авторской, известной), степень согласованности расчетных и экспериментальных данных.

Определяемые рецензентом: ошибочность методики или неправильность ее применения, наличие ошибки в расчетах или самой методике расчета, ошибочность в оценке гипотезы, подтверждение новой (автора) гипотезы, подтверждение известной гипотезы, степень согласованности расчетных и экспериментальных данных.

Наличие этих признаков для каждой работы позволит выполнить обширный анализ состояния научно-технических исследований в области мелиорации засоленных земель с использованием современных математических методов обработки информации на базе компьютерной техники. Подобный анализ позволит определить области с недостаточным уровнем исследований, выявить существующие проблемы и подготовить необходимые данные для прогноза научных исследований.

Знания продукционного типа представляются в виде:

ЕСЛИ v_1, v_2, \dots, v_N ТО д.

В данной форме v_1, v_2, \dots, v_N являются некоторыми высказываниями относительно объектов предметной области. Например, «средние многолетние суммарные осадки находятся в интервале 100 - 300 мм». В такого типа продукциях в случае истинности всех высказываний v_1, v_2, \dots, v_N выполняется действие «д», которое может заключаться в установлении некоторого факта. Например, «зона полупустынная». Действие «д» может также заключаться в выполнении определенной процедуры для получения значений атрибутов некоторого объекта, входящего в данную предметную область. Само правило также может быть представлено в виде определенной процедуры, разработанной для ЭВМ и реализующей тот или иной метод получения новых знаний и/или фактов на основе уже полученных ранее. В некоторых приложениях эти знания могут отображать перечень мероприятий и средств для решения проблем, обозначенных в соответствующей вершине дерева ДЕЙСТВИЯ в условиях, определенных в соответствующей вершине дерева ПРОДУКТА. Рассматривая многообразие областей применения данного подхода, необходимо отметить, что сами названия деревьев (ПРОДУКТ и ДЕЙСТВИЕ) достаточно условны. Например, изменив названия на СРЕДСТВА ЛЕЧЕНИЯ (аппаратура, лекарства и т.п.) и БОЛЕЗНЬ, мы переходим в совершенно иную область применения - медицину и т.д. Необходимо отметить тесную связь продукционных

знаний с наличием в базе знаний соответствующих научных трудов. При этом знания производного типа должны непосредственно вытекать из научных трудов, быть результатом научных исследований. С другой стороны, новые методы расчета должны оформляться в виде процедур на одном из языков программирования и пополнять процедурную часть базы знаний.

4. Структура базы знаний.

Логическая структура БЗ представлена в двух файлах древовидной структуры, содержащие категориально - понятийные деревья соответственно дерева ДЕЙСТВИЯ и дерева ПРОДУКТА. Структура этих файлов одинакова. Каждой вершине дерева соответствует одна запись. Первые 4 байта файла содержат максимальный идентификатор записи (вершины). Идентификатор присваивается каждой записи, является уникальным в файле и используется для различных ссылок на запись. В частности идентификатор записи идентифицирует различные объекты в других файлах, связанные с данной вершиной. Как будет показано ниже, каждой вершине соответствует список научных работ, идентификатор которого совпадает с идентификатором вершины. Записи файлов имеют следующую структуру. смещение длина содержимое

0	1	(n) Длина имени вершины.	
1	n	Имя вершины.	
n+1	4	Ссылка на вершину справа или 0.	
n+5	4	Ссылка на левую вершину нижнего уровня	или 0.
n+9	4	Идентификатор записи (списка рефератов).	

Последнее поле может использоваться как указатель на объекты в других файлах, содержащие, например, знания в виде продукций или программных модулей или других информационных объектов по желанию пользователя. Таким образом, последнее поле записи является интерфейсом для связи логической структуры БЗ с рабочей областью (базой данных). Очевидно, что таких рабочих областей может быть несколько, причем никаких ограничений на их структуру и количество нет.

В состав программного обеспечения входит модуль, предоставляющий пользователю удобные средства для корректировки структуры и содержимого обоих файлов. Это позволяет производить изменения в логической структуре БЗ не только на этапе ее разработки, но и на этапе эксплуатации. При этом защита знаний от разрушения осуществляется путем предупреждения и подробного объяснения возможных последствий таких действий. Логическая структура рабочей области БЗ, реализованной в данной версии системы состоит из трех файлов библиотечной структуры. Каждому дереву категориально-понятийного пространства (КПП) ставится в соответствие файл библиотечной структуры, содержащий списки идентификаторов научных трудов, ассоциированных с данной вершиной дерева. Идентификатором списка является идентификатор соответствующей вершины. Точка категориально-понятийного пространства или пара записей (вершин из деревьев) содержит два идентификатора, которым в библиотеках списков рефератов соответствуют два списка. Идентификаторы рефератов, содержащиеся одновременно в обоих списках, будут составлять список рефератов для данного элемента КПП. Необходимо отметить, что один и тот же идентификатор может содержаться в одном списке несколько раз. Это является следствием того, что данный элемент знаний ассоциирован одновременно с несколькими элементами КПП и этот список соответствует вершине, входящей во все или часть из этих элементов. Этот пример иллюстрируется элементом знаний с идентификатором «ИД 11».

Таким образом, наряду с двумя файлами древовидной структуры база знаний содержит еще три файла библиотечной структуры - две библиотеки списков научных трудов и библиотека, содержащая информацию (тексты) о научных трудах. Все три файла имеют

одинаковую структуру. Основной информационной единицей данных файлов является раздел, имеющий свой уникальный в пределах файла идентификатор.

Оглавление библиотечного файла является одним из разделов библиотеки и может находиться в любом ее месте. Отличие оглавления от остальных разделов заключается в том, что этот раздел не имеет своего идентификатора, а ссылка на него находится в описателе файла, находящегося в начале файла (первые шесть байт файла). Оглавление, как раздел библиотеки, состоит из элементов длиной 10 байт, содержащих необходимую для работы с разделом информацию: идентификатор раздела, длину раздела, ссылку на раздел (адрес относительно начала файла).

Библиотеки списков рефератов содержат разделы переменной длины. Каждый раздел является списком идентификаторов элементов знаний. Длина идентификатора равна 4-ем байтам. Количество идентификаторов определяется из длины раздела. Раздел библиотеки, содержащей собственно элемент знаний в виде реферата научной работы, должен быть разделен на фрагменты в соответствии с формулой исследования. При этом разделителем служит символ '\1' (код 01). Первый фрагмент раздела содержит атрибуты статьи (18 байт по одному на каждый атрибут) и ее название. Далее по формуле исследования.

Атрибуты реферата, заполняемые автором:

байт	значение
0	Наличие эксперимента
1	Использование известной методики эксперимента
2	Использование новой (автора) методики эксперимента
3	Подтверждение гипотезы экспериментом
4	Наличие известной теоретической методики расчета
5	Наличие новой (автора) теоретической методики расчета
6	Наличие известной эмпирической методики расчета
7	Наличие новой (автора) эмпирической методики расчета
8	Степень согласованности расчетных и эмпирических данных по десяти бальной шкале

Атрибуты реферата, заполняемые рецензентом:

байт	значение
9	Ошибочность методики
10	Неправильное применение методики
11	Ошибочность в оценке гипотезы
12	Подтверждение известной гипотезы
13	Подтверждение новой(автора) гипотезы
14	Степень согласованности расчетных и эмпирических данных по десяти бальной шкале
15-18	Резерв.

Такая структура БЗ позволяет проводить практически независимо друг от друга процессы построения логической структуры БЗ (категориально-понятийное пространство) и непосредственно наполнение базы конкретными знаниями. Ограничение здесь только одно - сначала необходимо создать элемент КПП и только после этого помещать в него знания. При любых изменениях в структуре КПП элемент знаний остается в БЗ (и может быть ассоциирован с любым другим элементом КПП) до тех пор, пока он ассоциирован хотя бы с одним элементом КПП, оставшимся в БЗ. Таким образом, сама проблема трансляции данных при изменении логической структуры отпадает.

5. Система управления базой знаний «КАРТА ЗНАНИЙ».

5.1 Общие сведения о программном комплексе.

Основными функциями системы управления базой знаний (СУБЗ) являются: построение, корректировка категориально-понятийного пространства (КПП), наполнение БЗ элементами знаний, учет доступа к информации, сервисные и другие вспомогательные функции.

5.2 Начало работы и парольная система.

В процессе инициации программного комплекса на экране появляется заставка с указанием названия комплекса, его разработчиков и необходимыми для обращения за справками реквизитами. Для продолжения работы комплекса необходимо нажать любой символ отличный от клавиши Esc. Нажатие клавиши Esc приводит к прекращению работы комплекса. В случае продолжения работы управление получает парольная система. На экране появляется окно со списком пользователей, имеющих доступ к БЗ. Используя клавиши управления курсором, пользователь должен подвести выделенную строку к своей фамилии и нажать клавишу Enter. Система запрашивает пароль. При вводе пароль на экране не отображается. Если пароль указан неверно, система предлагает продолжить сеанс в режиме демонстрации или завершить работу. Режим демонстрации отличается тем, что все изменения в БЗ либо совсем запрещены, либо имитируются для наглядности. В любом случае, информация, содержащаяся в БЗ, не изменяется. Если пароль указан правильно, на экране появляется окно главного меню со следующим списком основных функций системы:

- **Корректировка списка пользователей.**

Позволяет производить добавления и удаления пользователей, установку их статусов и паролей. Доступна пользователям со статусом 0.

- **Изменение собственного пароля.**

Пользователь с любым статусом может изменить свой пароль, но не статус.

- **Управление БЗ.**

Выполняет все функции, связанные с созданием и корректировкой КПП, и функции, связанные с заполнением БЗ.

- **Файл накопления фрагментов.**

При работе с информационными объектами предусмотрено выделение отдельных фрагментов информации и добавление их в специальный текстовый файл. В начале сеанса текущим файлом фрагментов является файл с именем DBKNFRAM.DOC. В процессе работы пользователю предоставляется возможность замены имени файла фрагментов. При этом старый файл сохраняется, если в него была произведена запись хотя бы одного фрагмента. Таким образом, пользователь может создавать набор файлов с фрагментами информации из БЗ для последующей обработки другими средствами.

- **Установка параметров БЗ.**

Основной особенностью СУБЗ, позволяющей существенно расширить область ее применения, является возможность корректировки ключевых параметров предметной области. Данная функция предназначена для изменения следующих параметров предметной области и функционирования системы: названия панелей, отображающих деревья КПП; имена файлов БЗ; название и структуру формулы исследования. Данная функция доступна пользователю со статусом 0.

- **Слияние баз знаний.**

Накопление знаний является наиболее трудоемким процессом в деятельности человека. Система предусматривает возможность коллективной работы с привлечением различных организаций, подразделений и специалистов для накопления знаний в данной предметной области. Данная функция предоставляет возможность слияния знаний из различных источников в единую интегрированную БЗ.

- Поиск по ключевым словам.

Если конкретное приложение предусматривает раздел формулы исследования «Ключевые слова», данная функция организует выборку элементов знаний по ключевым словам и (или) по атрибутам. Для задания условия выборки используются логические формулы по типу исчисления высказываний, термами которых являются ключевые слова и слова вида @n и #n, где n целое число, определяющее номер атрибута, а символы @ и # указывают соответственно на атрибут автора и эксперта. Другими словами в качестве запроса пользователь составляет логическую формулу, используя ключевые слова и операции &(И), |(ИЛИ) и !(НЕ) и круглые скобки. Далее происходит просмотр всех элементов знаний и выделяются те, для которых запрос является истинным. Наряду с этим функция поддерживает библиотеку запросов.

5.3 Создание логической структуры БЗ. Работа с деревьями КПП.

Выбор функции «Управление БЗ» приводит к появлению на экране двух панелей со списками вершин дерева ДЕЙСТВИЯ и дерева ПРОДУКТА. При этом текущей панелью является левая панель, соответствующая дереву ДЕЙСТВИЯ. Текущая панель определяется тем, что ее название отображается на красном фоне. Переключение текущей панели осуществляется нажатием клавиши Tab (табуляция). Для каждой панели фиксируется текущая вершина, имя которой отображается на светлом фоне. В данном фрагменте диалога пользователю предоставляются следующие возможности, выбор которых производится нажатием соответствующих функциональных клавиш.

- Завершение сеанса (Esc).
- Помощь (F1). На экране появляется окно с перечнем функциональных клавиш и действий, производимых системой при их нажатии.
- Вставка новой вершины (Insert). На экране появляется окно с выбором места расположения новой вершины - на том же уровне дерева или новая вершина должна стать подчиненной текущей. Затем пользователю предоставляется возможность ввести имя вершины.
- Удаление текущей вершины. Вершина удаляется только в том случае, если она терминальная, т.е. не имеет подчиненных вершин. При удалении вершины, информация, ассоциированная только с данной вершиной, удаляется из базы данных. В этом случае пользователь получает предупреждение, которое позволяет ему прекратить процесс удаления вершины или разрешить удаление. Информация, которая ассоциирована также и с другими вершинами остается в БЗ. Удаление вершины является логическим, т.е. удаление без возврата системе памяти, занимаемой удаленной вершиной. За счет этого, а также в результате корректировки названия вершины, если новое название не совпадает по длине со старым, в файлах деревьев появляются пустые места, что увеличивает размер файлов. Для удаления из файлов пустых мест и более экономного использования памяти на дисках необходимо произвести процедуру реорганизации БЗ.
- Корректировка имени вершины (F4). При нажатии данной клавиши в первой позиции названия текущей вершины появляется курсор. Используя клавиши управления курсором, удаления и клавиши символов, пользователь может изменить название вершины. Процесс корректировки заканчивается нажатием клавиш Esc или Enter. При этом клавиша Esc отменяет корректировку названия, а Enter приводит к изменению названия в БЗ. Никаких других изменений в БЗ не производится.
- Выбор элемента КПП (Enter). Система инициирует компоненту REFERAT.

На экране появляется список названий научных работ, соответствующих данному элементу КПП.

- Ввод последней выбранной работы в текущий элемент КПП (F2). Одна и та же научная работа может быть ассоциирована с несколькими элементами КПП. Данная функция необходима для дублирования (переноса) информации о научной работе из одного элемента КПП в другой. При этом сама эта информация содержится в БЗ в единственном экземпляре независимо от того, со сколькими элементами КПП она ассоциирована. Данный механизм работает следующим образом. Каждый раз при выборе научной работы из соответствующего списка в конкретном элементе КПП, компонента REFERAT запоминает необходимую информацию для выбранной работы. Таким образом, система всегда помнит последнюю научную работу, выбранную пользователем. При нажатии клавиши F2, система отображает на экране название последней выбранной научной работы с просьбой подтвердить ввод ее в текущий элемент КПП. В случае подтверждения информация о данной работе записывается в текущий элемент КПП. Если с начала сеанса не было выбрано ни одной работы, пользователь получает соответствующее сообщение.
- Отображение матрицы (F3). При нажатии клавиши F3 система инициирует компоненту DBKNMATR для отображения элементов КПП в виде категориально-понятийной матрицы, столбцы и строки которой соответствуют спискам вершин, расположенным на текущих уровнях соответствующих деревьев.

5.4. Матричное представление КПП.

В данной среде категориально-понятийное пространство представляется в виде ориентированного графа, вершинами которого являются категориально-понятийные матрицы (КПМ), а дуги отражают процессы детализации понятий. На экране появляется матрица, элементами которой являются квадраты, соответствующие элементам КПП. Квадраты имеют различную яркость, отражающую степень заполнения данного элемента КПП научными работами. Чем темнее квадрат, тем больше работ в соответствующем элементе КПП. Непосредственно под матрицей указаны номера и названия вершин текущего элемента КПМ и фактическое количество находящихся в нем научных работ. Изменение текущего элемента данной матрицы производится нажатием клавиш управления курсором (вверх, вниз, вправо, влево). Нажатие клавиши Esc приводит к возврату в среду работы с деревьями КПП. Для получения полного перечня функциональных клавиш, используемых в данной среде, необходимо нажать клавишу F1. В дополнении к выше перечисленным функциональным клавишам используются следующие клавиши:

Home - Переход к более высокому уровню в дереве ДЕЙСТВИЯ. На экране отображается матрица с теми же столбцами и строками, соответствующими вершинам дерева ДЕЙСТВИЯ, расположенным на один уровень выше. **PgUp** - Переход к более высокому уровню в дереве ПРОДУКТА. Аналогично клавише Home для дерева ПРОДУКТА.

End - Детализация понятия в дереве ДЕЙСТВИЯ. На экране отображается матрица с теми же столбцами и строками, соответствующими вершинам дерева ДЕЙСТВИЯ, являющимся подчиненными текущей вершине. **PgDn** - Детализация понятия в дереве ПРОДУКТА. Аналогично клавише End для дерева ПРОДУКТА.

Все выше перечисленные функциональные клавиши предоставляют пользователю полную возможность для отображения и качественного анализа КПМ любого уровня в графе категориально-понятийных матриц.

Нажатие клавиши Enter, как и в случае работы с деревьями КПП, приводит к переходу в среду ввода и корректировки информации о научных трудах, соответствующих текущему элементу КПП.

5.5. Ввод и корректировка элементов знаний.

Как было сказано выше, переход в среду ввода и корректировки информации о научных трудах осуществляется нажатием клавиши Enter после выбора текущей вершины КПП либо при работе с панелями деревьев, либо в режиме матричного представления КПП. При этом на экране отображается список названий научных трудов, ассоциированных с данным элементом КПП. Последняя строка в этом списке содержит текст 'Н О В Ы Й Э Л Е М Е Н Т'. Выбор этой строки указывает на необходимость ввода информации о научной работе, которой нет в данном элементе КПП. Выбор строки с названием научной работы устанавливает режим корректировки информации о данной работе. Выбор строки осуществляется клавишами управления курсором (вверх, вниз). При этом текущая строка отмечена горизонтальной стрелкой слева от первой строки названия работы. Для перехода к режиму ввода/корректировки необходимо нажать клавишу Enter. Информацию о научной работе можно удалить. При нажатии клавиши Del система предупреждает об удалении с требованием подтвердить решение пользователя об удалении. В случае подтверждения, работа удаляется из списка, ассоциированных с данным элементом КПП, и, если не найдено другого элемента КПП, с которым данная работа ассоциирована, информация о научной работе удаляется из базы знаний. При этом происходит логическое удаление. Это означает, что информация перестает быть доступной пользователям БЗ. Однако, физическое пространство памяти, которое эта информация занимала, остается в БЗ свободным до тех пор, пока не будет произведена процедура реорганизации базы знаний. Данное пространство так же может быть частично занято другой вновь поступившей информацией меньшего объема. Остаток пространства может быть возвращен системе только в результате реорганизации БЗ. Необходимо отметить, что подобная процедура происходит и при корректировке существующей информации. Если система установила, что информация была изменена, пользователь получает соответствующее предупреждение и после подтверждения система сначала удаляет старую информацию, затем добавляет измененную. Таким образом, пустые места могут появляться и в результате корректировки существующей информации. Клавиша Esc, как в большинстве других ситуаций, приводит к возврату в предыдущую среду.

После выбора названия научной работы система отображает список частей формулы исследования. Название формулы исследования, количество и названия ее частей содержатся в файле параметров системы (SETUP.KN) и могут быть изменены пользователем. При этом надо учитывать то, что уже существующая в БЗ информация соответствует предыдущей формуле. Пользователю необходимо следить за тем, чтобы либо изменения в формуле были незначительными, либо делать эти изменения при пустой базе знаний. Далее, работая с формулой исследования, пользователь выбирает нужную часть. Название текущей части формулы исследования отображается на светлом фоне. Выбор производится клавишами управления курсором (вверх, вниз). Подтверждение выбранной части осуществляется нажатием клавиши Enter. Клавиша Esc переводит систему в предыдущую среду (выбор названия работы). Все части формулы исследования, за исключением ниже отмеченных, обрабатываются одинаково. Происходит обращение к компоненте ТЕКСТОВЫЙ РЕДАКТОР. При этом появляется окно с названием данной части формулы в верхней рамке окна и текстом этой части, если он есть. Подробнее смотрите описание компоненты ТЕКСТОВЫЙ РЕДАКТОР. Для части 'Список пользователей' происходит то же самое, но, внесение каких либо изменений в эту часть запрещено. Для части 'Атрибуты работы' на экране отображается список

атрибутов, определяемых автором, и после нажатия клавиши Esc атрибутов, определяемых рецензентом. В обоих списках справа от названий атрибутов расположен столбец с их значениями. В этом столбце значение текущего атрибута отражено на светлом фоне. Значения бывают двух типов. Тип 'ДА/НЕТ' переключаются нажатием клавиши Enter. Числовые атрибуты устанавливаются путем ввода соответствующего числа.

Компонента REFERAT прекращает свою работу при нажатии клавиши Esc в среде выбора названия научной работы. При этом происходит возврат в среду, в которой она была вызвана. Это может быть среда выбора элемента КПП в режиме отображения деревьев или категориально-понятийных матриц.

5.6 Ввод и редактирование текста.

В тех случаях, когда необходимо ввести или отредактировать текст произвольной длины, система иницирует ТЕКСТОВЫЙ РЕДАКТОР. В среде редактора пользователю предоставляются широкие возможности для редактирования текста. Завершение работы с редактором используется клавиша Esc. Краткую информацию о работе редактора и используемых в его среде функциональных клавишах пользователь может получить, нажав клавишу F1. Процесс редактирования начинается в режиме вставки. Используя клавишу Insert, пользователь может изменить режим вставки на режим замены и обратно. В процессе редактирования, текст представляется как непрерывная последовательность символов. В режиме вставки текущий символ и все символы справа от него сдвигаются вправо, и очередной символ поступает на освободившееся место. В режиме замены очередной символ заменяет текущий. Это правило не относится к символу 'конец строки' см. ниже. Текущий режим обозначается словами 'вставка' и 'замена', отображаемыми в левой стороне нижней границы окна редактора вместе с номерами текущей строки и текущей позиции курсора. Разбиение текста на строки производится вставкой символа 'конец строки' в нужное место текста. Данный символ не отображается на экране. Его наличие определяется переходом текста на другую строку. Для вставки символа 'конец строки' необходимо нажать клавишу Enter в режиме вставки. Заметим, что в режиме замены клавиша Enter осуществляет перевод курсора в первую позицию следующей строки. Удаление символа 'конец строки' (слияние двух строк) производится так же, как и обычных символов, клавишами Del, если курсор находится в конце строки или клавишей <Del (удаление слева), если курсор находится в первой позиции следующей строки. В процессе редактирования пользователь может выделять блоки смежных символов в той или иной части текста. Одновременно может быть выделен только один блок. Выделение блока осуществляется клавишами управления курсором при нажатой клавише Shift. При выделении очередного блока отметка предыдущего блока гасится. В добавление к этому в среде редактора используются следующие функциональные клавиши.

- PgDn, PgUp - Листание текста на страницу (окно) вниз и вверх.
- Home, End - Перевод курсора в начало и конец текущей строки.
- Shift Del - Удаление блока.
- F2 - Сохранение текущего состояния текста в БЗ. В процессе редактирования текст постоянно находится в оперативной памяти ЭВМ, что приводит к потере результатов работы в случае аварийного или неправильного завершения работы. Данная функция позволяет фиксировать в рабочем файле промежуточные результаты редактирования с целью минимизировать потери в случае аварийного завершения.
- F3 - Загрузка текста из указанного файла. Пользователю предоставляются широкие возможности для поиска и указания необходимого файла см. компоненту ВЫБОР ФАЙЛА. Текст загружается в зависимости от режима. В режиме 'вставка' текст

вставляется в текущую позицию редактируемого текста. В режиме 'замена' текст полностью заменяет редактируемый.

- F4 - Загрузка текста, ранее зафиксированного при помощи клавиши F2.
- F5 - Добавление фрагмента в файл фрагментов. В качестве фрагмента выбирается весь раздел элемента знаний или блок, если он выделен.
- F6 - Печать фрагмента. В качестве фрагмента выбирается весь раздел элемента знаний или блок, если он выделен.
- F9 - Гашение выделенного блока. Информация сохраняется. Выделенного блока нет.

5.7 Определение имени файла.

Данная компонента инициируется везде, где необходимо указать имя файла, с которым будут произведены те или иные действия. На экране появляется окно, состоящее из двух частей. В верхней части расположена строка текущего каталога. В этой панели отображается путь к текущему каталогу и шаблон имени файла. Шаблон представляет собой имя файла, в котором могут встречаться символы '*' и/или '?'. При этом имя файла считается удовлетворяющим шаблону, если оно может быть получено заменой этих символов любыми допустимыми для имени файла символами. При замене знак "*" можно заменять на любое допустимое количество символов или опускать. Знак "?" заменяется на один и только на один символ. Переход в эту часть осуществляется клавишей PgUp. Пользователю предоставляется возможность отредактировать текст, находящийся в этой строке. После нажатия клавиши Enter система использует указанный файл, если указан путь к файлу. В противном случае осуществляется переход в нижнюю часть, где отображается список файлов и подкаталогов указанного каталога, удовлетворяющий данному шаблону. В данный список не включаются все файлы, имеющие расширения EXE, COM и OBJ. В нижней части, используя клавиши управления курсором, пользователь выбирает нужную строку и нажимает клавишу Enter. При этом производятся следующие действия. Если курсор отмечает имя файла, то выбирается этот файл. Если курсор отмечает имя каталога, то отображается содержимое данного каталога. Если курсор отмечает строку «..\», то отображается содержимое внешнего (содержащего данный) каталога. Завершение процесса выбора файла происходит либо при нажатии клавиши Esc, либо при явном указании файла в верхней или нижней частях окна и нажатии клавиши Enter. В любом случае текущий каталог операционной системы не меняется.

6. Структура программного обеспечения системы.

Система построена по модульному принципу с корневым управляющим модулем, инициирующим различные модули для выполнения соответствующих функций.

В настоящее время в систему входят следующие модули:

- MAPKN.EXE Модуль, вызываемый пользователем для инициации работы системы. Отображает начальную заставку. Затем переходит к процедуре ввода и проверки пароля и обработки списка пользователей системы. Модуль завершает работу вызовом модуля главного меню (DBKNMN.OVL).
- DBKN.OVL Модуль, выполняющий функции построения и корректировки логической структуры БЗ, а также заполнения последней.
- DBKNADD.OVL Слияние баз знаний.
- DBKNMN.OVL Модуль, находящийся в оперативной памяти на протяжении всего времени работы системы. Данный модуль управляет главным меню и выполняет

перечисленные в нем функции системы, динамически вызывая для этого соответствующие модули системы.

- DBKNSLKT.OVL Модуль поисковой системы. Обрабатывает запросы пользователя на выбор информационных объектов по атрибутам и ключевым словам.
- DBKNSTAT.OVL Отображает состояние файлов БЗ и с согласия пользователя реорганизует их для освобождения неиспользуемой памяти на дисках.
- EDITSTP.OVL Создание и редактирование файла параметров системы (SETUP.KN).

7. Структура базы знаний.

БЗ состоит из 6 файлов, имена которых указываются пользователем при создании (корректировке) файла параметров БЗ (SETUP.KN). В настоящем руководстве приведены условные имена этих файлов.

- TREE1.KN Файлы древовидной структуры, содержащие описание TREE2.KN логической структуры БЗ.
- LIST1.KN Файлы библиотечной структуры, содержащие списки LIST2.KN идентификаторов информационных объектов.
- LIBREF.KN Файл библиотечной структуры, предназначенный для хранения информационных объектов базы знаний.
- REQUEST.KN Файл библиотечной структуры, содержащий часто используемые поисковые запросы или их фрагменты.

В систему также входят 2 файла с фиксированными именами:

- SETUP.KN Файл параметров системы.
- USER.LIS Список пользователей системы.

В процессе работы могут быть созданы следующие файлы:

- DBKNSTAT.LST Отчет о состоянии файлов БЗ (модуль DBKNSTAT.OVL).
- DBKNEDIT.SAV Содержит последний сохраненный фрагмент текста при его редактировании.
- DBKNFRAM.DOC Файл накопления фрагментов информационных объектов (имя файла может быть изменено пользователем см. главное меню системы).

Заключение.

Система программ «КАРТА ЗНАНИЙ» построена по модульному принципу, что позволяет дальнейшее расширение функциональных возможностей системы. В дальнейшем предполагается разработать систему наполнения БЗ продукционными и функциональными знаниями. Это позволит перейти от накопления знаний к автоматизации процесса получения новых знаний.

Автоматизированная система "Карта знаний науки"

Разрабатываемая система "Карта знаний науки" создается как интегрированная база знаний, дающая общий методологический подход к обработке и представлению знаний в сфере сельскохозяйственного производства. Предлагаемый метод представления знаний дает возможность организовать быстрый поиск научно-технической информации за счет последовательной локализации области знаний.

I. Построение категориально-понятийного пространства.

Представление знаний с использованием аппарата категориально-понятийных матриц основано на построении двух деревьев - дерева ПРОДУКТА (П) и дерева ДЕЙСТВИЯ (Д). Процесс построения этих деревьев заключается в том, что каждая вершина соответствующего дерева обозначает некоторое понятие из соответствующей области действия или продукта системы категориально-понятийных матриц. При этом подчиненная вершина соответствует понятию, детализирующему понятие, соответствующее вершине более высокого уровня. Другими словами соответствующее родительской вершине понятие обобщает понятия, соответствующие подчиненным вершинам.

Далее рассмотрим декартово произведение деревьев П и Д как множество пар вида (p, d) , где p - вершина дерева П, а d - вершина дерева Д. Это произведение назовем категориально-понятийным пространством (КПП). В этом случае всякому отношению между вершинами деревьев П и Д будет соответствовать некоторое подмножество категориально-понятийного пространства.

Для построения представления знаний в области мелиорации засоленных земель целесообразно рассмотреть подпространство КПП, включающее в себя понятия из П и Д, входящие в указанную предметную область. При этом, подпространством КПП будем называть декартово произведение двух поддеревьев соответственно из П и Д, а суммой подпространств их простое объединение как подмножеств. Заметим очевидные свойства подпространств. Два подпространства либо не пересекаются, либо одно полностью содержится в другом. Это свойство непосредственно вытекает из аналогичного свойства поддеревьев. Другое важное свойство состоит в том, что подпространства не пересекаются если корневые вершины поддеревьев, являющихся проекциями этих подпространств на П или Д, хотя бы в одном из этих деревьев лежат на одном уровне. Сумма таких подпространств и будет в дальнейшем рассматриваться как предметная область и, в частности, система знаний в области мелиорации засоленных земель. Каждой точке (p, d) из КПП поставим в соответствие совокупность знаний о воздействии действия "д" на продукт "п" и назовем его элементом базы знаний $Z(p, d)$. Заметим, что $Z(p, d)$ может быть пуст, что отражает естественный факт отсутствия знания о воздействии действия "д" на продукт "п".

II. Структура элемента базы знаний.

Элемент базы знаний содержит информацию о научных трудах, содержащих знания о воздействии действия на продукт.

Научные труды представляются в базе знаний своими рефератами, составленными по определенным правилам в соответствии с формулой исследования, которая включает в себя следующие правила:

а) Дать четкое определение метатеории, которая является исходной по отношению к теории, развиваемой в исследовании. После слов "дополняющее известные знания", завершающих описание метатеории указать, чем существующая теория не отвечает требованиям практики, т.е. охарактеризовать цель исследования (неизвестное).

б) Описать, какие методы использовались для познания неизвестного.

в) Описать развиваемую теорию в конечном состоянии, т.е. в том виде, который она приобрела после проведения исследования.

г) Определить практическую значимость результатов, полученных за счет применения усовершенствованной теории.

д) Установить область возможного применения получившей развитие теории и получаемого при этом положительного эффекта.

Все научные труды, содержащиеся в базе знаний, должны сопровождаться следующими признаками.

- Установленные автором: наличие эксперимента, использование известной методики эксперимента, использование новой (автора) методики эксперимента, подтверждение гипотезы экспериментом или нет, наличие теоретической методики расчета (авторской, известной), наличие эмпирической методики расчета (авторской, известной), степень согласованности расчетных и экспериментальных данных.
- Установленные рецензентом: ошибочность методики или неправильность ее применения, наличие ошибки в расчетах или самой методике расчета, ошибочность в оценке гипотезы, подтверждение новой (автора) гипотезы, подтверждение известной гипотезы, степень согласованности расчетных и экспериментальных данных.

Наличие этих признаков для каждой работы позволит выполнить обширный анализ состояния научно-технических исследований в области мелиорации засоленных земель с использованием современных математических методов обработки информации на базе компьютерной техники. Подобный анализ позволит определить области с недостаточным уровнем исследований, выявить существующие проблемы и подготовить необходимые данные для прогноза научных исследований.

III. Описание программы.

Программа может эксплуатироваться в операционной системе MSDOS 2.0 и выше на ПЭВМ IBM XT/AT.

IV. Основные функции программы.

Программа работает в режиме диалога с пользователем. Управление осуществляется различными клавишами. Клавиша ESC приводит к прекращению работы в данном режиме и переход к предыдущему состоянию.

1. Работа с деревьями.

При вызове программы на экране монитора появляются две панели со списками вершин деревьев ДЕЙСТВИЯ и ПРОДУКТА. Уровень вершины в дереве отмечен сдвигом имени на соответствующее количество позиций вправо. Название панели (дерева) указано в верхней ее части.

При этом название текущей панели отображается на красном фоне. В каждой панели фиксируется текущая вершина, имя которой отображено черными буквами на белом фоне. Пара текущих вершин образует текущую точку категориально-понятийного пространства (КПП).

Для работы с панелями используются следующие клавиши:

1. ESC - Завершение сеанса работы;
2. F1 - Отображение списка функциональных клавиш с краткими пояснениями;
3. TAB - Клавиша табуляции переключает панели;
4. Ins - Добавление новой вершины;
5. Del - Удаление текущей вершины;
6. F2 - Ввод в данную точку КПП последнего выбранного или введенного реферата;
7. Enter - Получение списка рефератов в данной точке КПП для выбора или ввода нового реферата.

При переходе в режим добавления новой вершины на экране в области пассивной панели появляется меню, состоящее из трех строк - отмена создания новой вершины, создание вершины на текущем уровне (справа от данной вершины), создание вершины на нижнем уровне (слева от существующих). Используя Enter, пользователь выбирает необходимое действие. В случае выбора одного из двух вариантов создания новой вершины, программа запрашивает имя новой вершины, предоставляя пользователю окно для ввода этого имени. В любой момент времени клавиша ESC отменяет режим создания новой вершины.

При успешном вводе имени вершины, оно включается в список имен данной панели (дерева) на соответствующем уровне.

2. Работа со списком рефератов.

В режиме списка рефератов пользователь может просмотреть этот список, выбрать нажатием клавиши Enter нужный реферат для просмотра и (или) корректировки. Для ввода нового реферата необходимо выбрать строку 'Новый реферат'.

3. Ввод нового реферата.

Ввод нового реферата осуществляется в соответствии с формулой исследования. Для этого необходимо:

- Определить исходные понятия, воздействующие на обсуждаемые в работе процессы и явления.
- Установить в дереве ДЕЙСТВИЯ соответствующую текущую вершину, создав при необходимости новую.
- Определить понятия, являющиеся результатом воздействия исходных понятий.
- Установить в дереве ПРОДУКТА соответствующую текущую вершину, создав при необходимости новую.
- Нажатием клавиши Enter, перейти в режим списка рефератов для данной точки КПП.
- Выбрать строку 'Новый реферат' и нажать клавишу Enter для ввода нового реферата. При этом на экране появится список разделов реферата в соответствии с формулой исследования. Используйте клавишу Enter для выбора нужного раздела. Необходимым является только раздел, содержащий название статьи. Остальные разделы вводятся по желанию пользователя.

После ввода реферата необходимо в режиме работы с панелями (деревьями) ввести данный реферат в остальные точки КПП, к которым он принадлежит. Для этого необходимо перейти в режим работы с деревьями, нажимая клавишу ESC несколько раз пока на экране не появятся две панели со списками вершин деревьев. После этого, отметив соответствующие вершины в каждом дереве, ввести данный реферат, используя функциональную клавишу F2.

Разработка базы знаний с использованием аппарата категориально-понятийного пространства.

Носителем научного потенциала является научно-техническая информация. Действующие ныне принципы организации обработки НТИ были заложены более 30 лет назад. За это время потоки НТИ и объемы информации возросли более чем в 5 раз. При таких темпах накопления информации прежние способы обработки НТИ не могут удовлетворить потребностей в доступе к новым знаниям. В настоящее время уровень вовлечения новых знаний в оборот научно-производственной деятельности составляет около 2%.

Появившаяся новая техника и новые информационные технологии могут и должны быть применены к способам обработки НТИ. Принципиальной новизной современных компьютерных технологий является ориентация на обработку знаний. Для обработки НТИ это означает необходимость разработки принципов и способов представления содержательной, смысловой информации научно-технического документа, а не только его библиографических реквизитов. Отображением смысловой информации, содержащейся в документе, в настоящее время считается составляемый автором реферат. Такое представление является субъективным и не всегда отражает реальную ценность работы. Автореферирование должно осуществляться на общих принципах представления знаний, принятых в данной предметной области.

Разрабатываемая система "Карта знаний науки" создается как интегрированная база знаний, дающая общий методологический подход к обработке и представлению знаний. Предлагаемый метод представления знаний дает возможность организовать быстрый поиск научно-технической информации за счет последовательной локализации области знаний. Эксперименты, проведенные в Московском Гидромелиоративном институте с группами студентов и аспирантов, показали, что поиск литературы по заданной тематике ускоряется в 200- 300 раз по сравнению с традиционными. Автоматизация интеллектуального труда человека за счет внедрения данного метода на современных ЭВМ еще больше ускорит поиск и подбор литературы по заданной теме, предоставив возможность человеку больше времени уделить непосредственно творческому процессу.

1. Построение категориально-понятийного пространства.

Представление знаний с использованием аппарата категориально-понятийных матриц основано на построении двух деревьев - дерева ПРОДУКТА (П) и дерева ДЕЙСТВИЯ (Д). Процесс построения этих деревьев заключается в том, что каждая вершина соответствующего дерева обозначает некоторое понятие из соответствующей области действия или продукта системы категориально-понятийных матриц. При этом подчиненная вершина соответствует понятию, детализирующему понятие, соответствующее вершине более высокого уровня.

Другими словами соответствующее родительской вершине понятие обобщает понятия, соответствующие подчиненным вершинам.

Далее рассмотрим декартово произведение деревьев П и Д как множество пар вида (п, д), где п - вершина дерева П, а д - вершина дерева Д. Это произведение назовем категориально-понятийным пространством (КПП). В этом случае всякому отношению между вершинами деревьев П и Д будет соответствовать некоторое подмножество категориально-понятийного пространства.

Для построения представления знаний в предметной области целесообразно рассмотреть подпространство КПП, включающее в себя понятия из П и Д, входящие в указанную предметную область. При этом, подпространством КПП будем называть декартово произведение двух поддеревьев соответственно из П и Д, а суммой подпространств их простое объединение как подмножеств. Заметим очевидные свойства подпространств. Два подпространства либо не пересекаются, либо одно полностью содержится в другом. Это свойство непосредственно вытекает из аналогичного свойства поддеревьев. Другое важное свойство состоит в том, что подпространства не пересекаются если корневые вершины поддеревьев, являющихся проекциями этих подпространств на П или Д, хотя бы в одном из этих деревьев лежат на одном уровне. Сумма таких подпространств и будет в дальнейшем рассматриваться как предметная область.

Каждой точке (п,д) из КПП поставим в соответствие совокупность знаний о воздействии действия "д" на продукт "п" и назовем его элементом базы знаний $Z(p,d)$. Заметим, что $Z(p,d)$ может быть пуст, что отражает естественный факт отсутствия знания о воздействии действия "д" на продукт "п".

II. Структура элемента базы знаний.

Элемент базы знаний содержит два типа знаний - информацию о научных трудах, содержащих знания о воздействии действия на продукт и знания продукционного типа, позволяющие на основе существующих фактов получить новое знание или проверить выдвигаемые гипотезы.

Научные труды представляются в базе знаний своими рефератами, составленными по определенным правилам в соответствии с формулой исследования, которая включает в себя следующие правила:

а) Дать четкое определение метатеории, которая является исходной по отношению к теории, развиваемой в исследовании. После слов "дополняющее известные знания", завершающих описание метатеории указать, чем существующая теория не отвечает требованиям практики, т.е. охарактеризовать цель исследования (неизвестное).

б) Описать, какие методы использовались для познания неизвестного.

в) Описать развиваемую теорию в конечном состоянии, т.е. в том виде, который она приобрела после проведения исследования.

г) Определить практическую значимость результатов, полученных за счет применения усовершенствованной теории.

д) Установить область возможного применения получившей развитие теории и получаемого при этом положительного эффекта.

Все научные труды, содержащиеся в базе знаний, должны сопровождаться следующими признаками.

○ Установленные автором: наличие эксперимента, использование известной методики эксперимента, использование новой (автора) методики эксперимента, подтверждение гипотезы экспериментом или нет, наличие теоретической методики расчета (авторской, известной), наличие

эмпирической методики расчета (авторской, известной), степень согласованности расчетных и экспериментальных данных.

○ Установленные рецензентом: ошибочность методики или неправильность ее применения, наличие ошибки в расчетах или самой методике расчета, ошибочность в оценке гипотезы, подтверждение новой (автора) гипотезы, подтверждение известной гипотезы, степень согласованности расчетных и экспериментальных данных.

Наличие этих признаков для каждой работы позволит выполнить обширный анализ состояния научно-технических исследований в предметной области с использованием современных математических методов обработки информации на базе компьютерной техники. Подобный анализ позволит определить области с недостаточным уровнем исследований, выявить существующие проблемы и подготовить необходимые данные для прогноза научных исследований.

Знания продукционного типа представляются в виде:

ЕСЛИ v_1, v_2, \dots, v_N ТО d .

В данной форме v_1, v_2, \dots, v_N являются некоторыми высказываниями относительно объектов предметной области. Действие " d " может также заключаться в выполнении определенной процедуры для получения значений атрибутов некоторого объекта, входящего в данную предметную область.

Необходимо отметить тесную связь продукционных знаний с наличием в базе знаний соответствующих научных трудов. При этом знания продукционного типа должны непосредственно вытекать из научных трудов, быть результатом научных исследований. С другой стороны, новые методы расчета должны оформляться в виде процедур на одном из языков программирования и пополнять процедурную часть базы знаний.

ПОЛЬЗОВАТЕЛЮ АВТОМАТИЗИРОВАННОЙ СИСТЕМОЙ КАРТА ЗНАНИЙ

В проблеме развития мыслительности, кроме гениальности мыслителя, представляется целесообразным иметь как бы помощника в виде "карты знаний", которая представляла бы мыслителю, как бы с "птичьего полета" возможность осматривать имеющиеся "придумки" в сжатом, выделенном, формализованно обозначенном виде (на подобие карт географических, почвенных, геологических и т.д.), т.е. некоторых моделях действительности и абстракций.

Нам представляется, что методологической и методической основой могут являться концепция пространства категориально-понятийных матриц, связанных с созданием различных продуктов (результатов деятельности).

Категориально-понятийная матрица имеет следующий вид:

Действие	Продукт (результат)				
	Вещественный	Энергетический	Социальный	Экономический	Интеллектуальный
Человек	1.1	1.2	1.3	1.4	1.5
Среда	2.1	2.2	2.3	2.4	2.5
Инженерная система	3.1	3.2	3.3	3.4	3.5
Материал	4.1	4.2	4.3	4.4	4.5
Информация	5.1	5.2	5.3	5.4	5.5
Модель	6.1	6.2	6.3	6.4	6.5
Время	7.1	7.2	7.3	7.4	7.5
Управление	8.1	8.2	8.3	8.4	8.5

Продукт (в широком смысле) является функцией восьми компонент:

$P = \Phi(\text{Чел.}, \text{Среда}, \text{Инж.сист.}, \text{Матер.}, \text{Инф.}, \text{Модель}, \text{Время}, \text{Упр.})$.

Каждая клеточка категориально-понятийной матрицы является неким элементарным процессом. Например: 1.1 - влияние человека на получение вещественного продукта и т.д.

При этом каждую компоненту можно рассматривать с двух позиций: с физической (материальной) и абстрактной, например, человек с его физическими характеристиками или его формализованное представление. Таким образом, мы имеем два категориально-понятийных дерева: дерево действия и дерево продукта, которые могут иметь понятия следующих уровней, например: среда: политическая, социальная, экономическая, техническая, природная, а, например, природная среда: климатические, гидрологические, почвенные, гидрогеологические, инженерно-геологические, биологические условия. В свою очередь, например, климатические условия, описываются: радиацией, осадками и т.д.

Аналогично можно расписать любое из понятий, приведенных в категориально-понятийной матрице, т.е. построить (для данной предметной области) категориально-понятийное дерево.

Построив, таким образом, два дерева: ДЕЙСТВИЕ и ПРОДУКТ, можно переходить к рассмотрению множества пар вершин обоих деревьев.

Такое множество называется категориально-понятийным пространством и, по нашему мнению, является удобным объектом для построения карты знаний. В данном случае знания, в той или

иной форме отражающие влияние определенного действия на некоторый продукт, помещаются в соответствующую точку категориально-понятийного пространства, что и составляет базу знаний о данной предметной области.

Чтобы в этом пространстве разместить (в формализованном представлении) мысли, излагаемые в фразе, статье и других публикациях, для этого пользователю предоставляется возможность с помощью программной оболочки для ПЭВМ создать пространство категориально-понятийных матриц и разместить в нем любое знание о процессе и (или) деятельности (описание программной оболочки содержится в подсказках внутри ПЭВМ). Для этого Вам необходимо осмыслить или то, что Вы узнали или придумали о процессе, или как Вы его представляете через категории, общенаучные понятия или "просто" слова. При этом, вначале надо их разделить на две группы ключевых слов: в одну группу отнести слова, которые обозначают содержание продукта (результата) процесса или деятельности, во вторую аналогично.

Например, формулируется мысль (в явном или скрытом виде): "В эксперименте установлено, что влажность почвы зависит от исходного увлажнения почвы, режима подачи воды дождевальной техникой при росте и развитии сельскохозяйственных растений во времени".

В этом примере "влажность почвы" является результатом процесса или деятельности, а "исходное увлажнение почвы", "вода", "дождевальная техника", "растение" и "время", есть компоненты, от которых зависит "влажность почвы". От режима подачи, роста и развития тоже зависит процесс создания "влажности почвы", но эти ключевые слова отражают (описывают, означают) сопутствующие (внутренние) процессы, т.е. процессы другого уровня и их выносить в ранг ключевых слов не надо. После осмысления вносимых в ПЭВМ знаний необходимо:

1. Дискету, содержащую описание программной оболочки, переписать на жесткий диск и сделать директорий с системой текущим. Для этого выполнить следующую последовательность команд MS-DOS:

```
md dbkn - создание директория для системы;
cd dbkn - директорий DBKN - текущий;
copy a:\*.* c:\dbkn - копирование файлов с дискеты.
```

2. Запустить систему, выполнив команду MAPKN, при этом должна появиться заставка, что свидетельствует о нормальном запуске.

3. Для определения содержимого базы знаний удобней воспользоваться матричным представлением категориально-понятийного пространства. Для этого, при появлении двух панелей со списками вершин, необходимо нажать F3 и в дальнейшем пользоваться подсказкой, используя клавишу F1.

4. Остальные соображения об описании и характеристике знаний от Вас "просит" ПЭВМ.

Примечания.

1. Длина названия работы не более 400 символов и количество строк не более 5.

2. Количество работ в точке КПП не более 150.

3. Длина текста работы не более 10000 символов.

4. Длина текста всех разделов "Формулы исследования" не более 10000 символов.

Для удаления реферата из базы знаний необходимо в окне "Список работ" для данной работы использовать клавишу [DEL] (см подсказку в ниж. строке).

*** Текстовый редактор ***

СПИСОК ФУНКЦИОНАЛЬНЫХ КЛАВИШ

Клавиши управления курсором (при нажатой клавиши Shift происходит выделение и стирание блока)

_	- Шаг на строку вверх		Home	- Переход в начало строки
_	- Шаг на строку вниз		End	- Переход в конец строки
->	- Шаг на символ вправо		PgUp	- Шаг на страницу вверх

<- - Шаг на символ влево | PgDn - Шаг на страницу вниз

-----+-----

Ins - Переключение режимов 'вставка'/'замена'

F9 - Удаляет блок (текст сохраняется)

Del - Удаление символа, отмеченного курсором

ŷDel - Удаление символа слева от курсора

Enter - В режиме 'вставка' вставляет символ 'конец строки' В режиме 'замена' переводит курсор в начало следующей строки

F2 - Сохранение текста.

F3 - Загрузка нового файла или его фрагмента.

В режиме 'ВСТАВКА' - вставляет новый текст в текущую позицию курсора.

В режиме 'ЗАМЕНА' - заменяет существующий текст.

При нормальном завершении файл запоминается и может быть загружен при помощи комбинации Shift+F3 без указания имени.

F4 - Загрузка ранее сохраненного функцией F2 текста. (Аналогично F3)

F5 - Добавление фрагмента к файлу фрагментов.

В качестве фрагмента выбирается выделенный блок или весь раздел, если блок невыделен. Используйте F9 для сброса выделенного блока.

F6 - Печать фрагмента. Аналогично F5.

F9 - Удаляет выделение блока (текст сохраняется).

*** Программа выбора файла ***

Клавиши управления курсором

_ - Шаг на строку вверх

_ - Шаг на строку вниз

-> - Шаг на столбец вправо или в конец списка

<- - Шаг на столбец влево или в начало списка

Home - Переход в начало списка файлов

End - Переход в конец списка файлов

PgUp - Изменение строки шаблона или явное указание имени файла

Клавиша Enter

Если курсор отмечает имя файла, то выбирается этот файл.

Если курсор отмечает имя каталога, то отображается содержимое данного каталога.

Если курсор отмечает строку, то отображается содержимое каталога, содержащего данный.

Если курсор находится в строке шаблона, то отображается новый список в соответствии с шаблоном или выбирается явно заданный файл.

*** Модуль TRANTREE ***

ИЗМЕНЕНИЕ НАЗВАНИЙ ВЕРШИН

Левая панель содержит названия вершин, которые были до начала выполнения функции, и не изменяется. Правая панель служит для внесения изменений в названия вершин. Первоначально правая панель полностью совпадает с левой.

Список функциональных клавиш:

Ctrl D - Стирание названия текущей вершины.

<-- --> - Движение курсора в поле названия

Home - Курсор в начало поля названия

End - Курсор в конец поля названия

__ - Движение по дереву.

Del - Стирание символа

Backspace - Стирание символа слева от курсора

*** ПАНЕЛИ ДЕРЕВЬЕВ (DBKN) ***

Ins - Вставка новой вершины

Del - Удаление текущей вершины

__ - Движение по дереву

Enter - Выбор элемента КПП

Tab - Выбор дерева

F2 - Ввод последней выбранной работы в текущий элемент

КПП F3 - Отображение матрицы

F4 - Корректировка названия вершины

*** ОТОБРАЖЕНИЕ МАТРИЦЫ (DBKNMATR) ***

Список функциональных клавиш:

Home - Переход к родительской вершине дерева 'ДЕЙСТВИЯ'

PgUp - Переход к родительской вершине дерева 'ПРОДУКТА'

End - Переход к подчиненной вершине дерева 'ДЕЙСТВИЯ'

PgDn - Переход к подчиненной вершине дерева 'ПРОДУКТА'

--> <-- __ - Движение по матрице

Enter - Список элементов знаний

Esc - Отображение панелей

*** СОСТОЯНИЕ БЗ (DBKNSTAT) ***

Количество вершин - количество понятий (категорий), содержащихся в данном дереве КПП.

Количество списков - соответствует количеству вершин дерева КПП, содержащих информационные объекты БЗ.

Длина файла - объем памяти в байтах, занимаемый данным файлом на диске.

Степень заполнения - определяет долю полезной информации в файле. Эта доля уменьшается в результате корректировки данного файла. При уменьшении показателя до определенного порога система предлагает реорганизовать данный файл для освобождения неиспользуемой памяти.

Количество разделов - количество информационных объектов базы знаний.

*** Просмотр файла (DBKNVIEW) ***

СПИСОК ФУНКЦИОНАЛЬНЫХ КЛАВИШ

Клавиши управления курсором (при нажатой клавиши Shift происходит выделение и стирание блока)

_ - Шаг на строку вверх		Home - Переход в начало строки
_ - Шаг на строку вниз		End - Переход в конец строки
-> - Шаг на символ вправо		PgUp - Шаг на страницу вверх
<- - Шаг на символ влево		PgDn - Шаг на страницу вниз

-----+-----

Esc - Отмена вставки информации из данного файла.

Enter - Вставка выделенного блока или всего файла, если блок не выделен.

F9 - Удаляет выделение блока (текст сохраняется).

*** Подсказки главного меню (10 - 29) ***

Управление базой знаний

Выполняет все функции, связанные с созданием и корректировкой КПП, и функции, связанные с заполнением БЗ.

Файл накопления фрагментов.

При работе с информационными объектами предусмотрено выделение отдельных фрагментов информации и добавление их в специальный текстовый файл. В начале сеанса текущим файлом фрагментов является файл с именем DBKNFRAM.DOC. В процессе работы пользователю предоставляется возможность замены имени файла фрагментов. При этом старый файл сохраняется, если в него была произведена запись хотя бы одного фрагмента. Таким образом, пользователь может создавать набор файлов с фрагментами информации из БЗ для последующей обработки другими средствами.

Установка параметров БЗ.

Основной особенностью СУБЗ, позволяющей существенно расширить область ее применения, является возможность корректировки ключевых параметров предметной области. Данная функция предназначена для изменения следующих параметров предметной области и функционирования системы: названия панелей, отображающих деревья КПП; имена файлов БЗ; название и структуру формулы исследования. Данная функция доступна пользователю со статусом 0.

Реорганизация базы знаний

Удаление вершины или элемента знаний является логическим, т.е. удаление без возврата системе памяти, занимаемой удаленной вершиной. За счет этого, а также в результате корректировки названия вершины, если новое название не совпадает по длине со старым, в файлах деревьев и/или информационных объектов появляются пустые места, что увеличивает размер файлов. Для удаления из файлов пустых мест и более экономного использования памяти на дисках необходимо произвести процедуру реорганизации БЗ. При этом пользователю предоставляется статистическая информация о состоянии файлов системы и предложения о реорганизации нерационально заполненных файлов.

Слияние баз знаний.

Накопление знаний является наиболее трудоемким процессом в деятельности человека. Система предусматривает возможность коллективной работы с привлечением различных организаций, подразделений и специалистов для накопления знаний в данной предметной области. Данная функция предоставляет возможность слияния знаний из различных источников в единую интегрированную БЗ.

Поиск по ключевым словам.

Если конкретное приложение предусматривает раздел формулы исследования "Ключевые слова", данная функция организует выборку элементов знаний по ключевым словам и(или) по атрибутам. Для задания условия выборки используются логические формулы по типу исчисления высказываний, терминами которых являются ключевые слова и слова вида @n и #n, где n целое число, определяющее номер атрибута, а символы @ и # указывают соответственно на атрибут автора и эксперта. Другими словами в качестве запроса пользователь составляет логическую формулу, используя ключевые слова и операции &(И), |(ИЛИ) и !(НЕ) и круглые скобки. Далее происходит просмотр всех элементов знаний, и выделяются те, для которых запрос является истинным. Наряду с этим функция поддерживает библиотеку запросов.^#&

 ДЕМОНСТРАЦИОННЫЕ МАТЕРИАЛЫ (БЕЗ КОМЕНТАРИЕВ)
СПИСОК ПОЛЬЗОВАТЕЛЕЙ

Китаев Ю.А. Рекс Л.М.

ESC - Конец ↑ ↓ Enter - Выбор

Управление базой знаний

Файл накопления фрагментов

Установка параметров базы знаний

Реорганизация базы знаний

Слияние баз знаний

Поиск по ключевым словам

F1 - Помощь Используйте ↑ ↓ Enter для указания функции ESC - Конец

Файл накопления фрагментов

Имя файла(путь) для сбора фрагментов текстов
--

DBKNFRAM.DOC

ESC - Отмена

Управление базой знаний

Файл накопления фрагментов

Установка параметров базы знаний

Реорганизация базы знаний

Слияние баз знаний

Поиск по ключевым словам

F1 - Помощь Используйте ↑ ↓ Enter для указания функции ESC - Конец

Состояние базы знаний на 13 февраля 2004

Дерево Д Е Й С Т В И Е

Дерево П Р О Д У К Т

Количество вершин	20	Количество вершин	20
Количество списков	2	Количество списков	4
Количество пустых вершин	18	Количество пустых вершин	16
Длина файла дерева	485	Длина файла дерева	469
Степень заполнения	100.0%	Степень заполнения	100.0%
Длина файла списков	112	Длина файла списков	144
Степень заполнения	44.6%	Степень заполнения	51.3%

 Длина файла информационных разделов 42145

Степень заполнения 95.6%

Количество разделов 6

F1 - Подсказка ESC-Конец F3-Печать DEL-Удаление

Управление базой знаний

Файл накопления фрагментов

Установка параметров базы знаний

Реорганизация базы знаний

Слияние баз знаний

Поиск по ключевым словам

F1 - Помощь Используйте ↑ ↓ Enter для указания функции ESC - Конец

Список запросов

сист
1989
мысль
all

Esc - Гл. меню ↑ ↓ Enter - Выбор Ins - Новый F10 - Коррект. Del - Удал.

ДЕЙСТВИЕ	ПРОДУКТ
Действие	Продукт
Среда	Вещественный
Политическая	Водный
Социальная	Солевой
Техническая	Пищевой
Экономическая	Сельхоз культуры
Природная	Окружающая среда
Человек	Энергетический
Материал	Тепловой
Информация	Социальный
Модель	Экономический
Время	Стоимость
Управление	Качество
Инженелная система	Производительность
Сельхозпроизводство	Интеллектуальный
Растеневодство	Фраза
Животноводство	Методология
Гидромелиоративная	Концепция
Оросительная	Методика
Осушительная	Мысль

F1 - Помощь ESC - Выход Enter - Выбор элемента КПП

	1	2	3	4	5
1	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺
2	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺
3	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺
4	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺
5	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺
6	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺	☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺

(2/4) Информация / Методология
4 Элементов знаний

F1 - Помощь ESC - Выход

F1 - Помощь

Ins - Вставка новой вершины
Del - Удаление текущей вершины
↑ ↓ - Движение по дереву
Enter - Выбор элемента КПП
Tab - Выбор дерева
F2 - Ввод последней выбранной
○ работы в текущий элемент КПП
F3 - Отображение матрицы
F4 - Корректировка названия вершины

ESC - Выход

Список элементов знаний

Рекс Л.М., Китаев Ю.А. Памятка пользователю автоматизированной системы
карта знаний

ВОПРОСЫ ЭКОНОМИЧЕСКОЙ УСТОЙЧИВОСТИ ВОДОХОЗЯЙСТВЕННЫХ СИСТЕМ

БОСТАНЖОГЛО М.М., ШНЯКИНА Л.Н.

ВОПРОСЫ ГИДРОТЕХНИЧЕСКОГО СТРОИТЕЛЬСТВА В ГОРНЫХ УСЛОВИЯХ .- ТЕЗИСЫ

ДОКЛАДОВ .-

КОБУЛЕТИ 1989 .- С.88

СТРАНА: СССР ЯЗЫК ДОК-ТА: РУССКИЙ

ОРГАН-ХРАНИТЕЛЬ: ЦНТИ, 15439

ОЦЕНКА ПЛОДородия МЕЛИОРИРОВАННЫХ СОЛОНЦОВЫХ КОМПЛЕКСОВ В

ДОННИКОВОМ АГРОЦЕНОЗЕ БУРЛАКОВА Л.М., ОЖГИБИЦЕВА Е.Я.

СИБИРСКИЙ ВЕСТНИК СЕЛЬСКОХОЗЯЙСТВЕННОЙ НАУКИ .- ЖУРНАЛ .

- 1989 .- №3 .- С.72-78

СТРАНА: СССР ЯЗЫК ДОК-ТА: РУССКИЙ

ОРГАН-ХРАНИТЕЛЬ: ЦНТИ

---> Н О В Ы Й Э Л Е М Е Н Т

↑ ↓ Enter - Выбор DEL - Удаление ESC - Выход

Формула исследования

Название работы
Содержание работы
Сведения об авторах
Список ключевых слов и фраз
Определение исходной теории
Методы познания неизвестного
Конечное состояние развиваемой теории
Практическая значимость результатов
Область применения результатов исследования
Список пользователей
Атрибуты работы

↑ ↓ Enter - Выбор раздела ESC – Выход

Атрибуты статьи, заполняемые автором(читателем)

- | | | |
|--|-----|-----------|
| 1. Наличие эксперимента | НЕТ | ↑ ↓ Enter |
| 2. Использование известной методики | НЕТ | |
| 3. Использование новой (автора) методики | ДА | |
| 4. Подтверждение гипотезы экспериментом | ДА | |
| 5. Наличие известной теоретической методики расчета | НЕТ | |
| 6. Наличие новой теоретической методики расчета | ДА | |
| 7. Наличие известной эмпирической методики расчета | НЕТ | |
| 8. Наличие новой эмпирической методики расчета | НЕТ | |
| 9. Степень согласованности расч. и эксп. данных (по десятибальной системе) | 10 | |

ESC – ВЫХОД

Атрибуты статьи, заполняемые рецензентом

- | | | |
|--|-----|-----------|
| 1. Ошибочность методики | НЕТ | ↑ ↓ Enter |
| 2. Неправильное применение методики | НЕТ | |
| 3. Ошибка в оценке гипотезы | НЕТ | |
| 4. Подтверждение известной гипотезы | НЕТ | |
| 5. Степень согласованности расч. и эксп. данных (по десятибальной системе) | 0 | |

ESC - ВЫХОД

ДЕЙСТВИЕ	ПРОДУКТ
Действие Среда Политическая Социальная Техническая Экономическая Природная Климатические Человек Материал Информация Модель Управление Инженерная система Сельхозпроизводство Растениеводство Животноводство Гидромелиоративная Оросительная	<div style="border: 1px solid black; padding: 5px; margin-top: 20px;"> <p>Отмена создания вершины</p> <p>Вершина на текущем уровне</p> <p>Вершина на нижнем уровне</p> </div>

F1 - Помощь ESC - Выход Enter - Выбор элемента КПП

Корректировка файла параметров

ЛЕВАЯ ПАНЕЛЬ

Название	ДЕЙСТВИЕ
Имя корневой вершины	Действие
Имя файла дерева	tree1.kn
Имя файла библиотеки списков	list1.kn

ПРАВАЯ ПАНЕЛЬ

Название	ПРОДУКТ
Имя корневой вершины	Продукт
Имя файла дерева	tree2.kn
Имя файла библиотеки списков	list2.kn
Имя файла библиотеки разделов	libref.kn
Имя файла библиотеки запросов	libreq.kn
Количество строк в окне панели	20
Количество частей формулы иссл.	11
Номер части раздела 'Ключевые слова'	4
Номер части раздела 'Атрибуты'	11
Формула исследования	↑ ↓ Enter

ESC – Конец

Файл списков дерева **ДЕЙСТВИЕ** используется меньше, чем на 75%.
Файл реорганизовать ?

ДА НЕТ

Используйте клавиши → ← Enter для выбора ответа

ВЫБОР ФАЙЛА

D:\DOCUME~1\LEONID~1\РАБОЧИ~1\LEK-MGUP\IDEI

..\

libref kn

libreq kn

list1 kn

list2 kn

setup kn

tree1 kn

tree2 kn

Enter - Выход из каталога. ESC - Выход.

PgUp - Редактирование строки шаблона.

F1 - Помощь

Выберите файл параметров исходной БЗ

HTML

1. Вводные замечания

- Для чего нужно это руководство?
- Что такое HTML?
- Что Вам понадобится для освоения HTML?
- Необходимое отступление о редакторах HTML

2. Наш первый HTML-документ

- Как устроен HTML-документ
- Обязательные метки
 - `<html>` и `</html>`
 - `<head>` и `</head>`
 - `<title>` и `</title>`
 - `<body>` и `</body>`
 - `<H1>` и `</H1>` ‡ `<H6>` и `</H6>`
 - `<P>` и `</P>`

3. Внутри абзаца

- Непарные метки
 - Конец строки: `
`
 - Горизонтальная линия: `<HR>`
 - &-последовательности
 - Комментарии
- Форматирование шрифта
 - Физические стили
 - Логические стили

4. Организация текста внутри документа

- Ненумерованные списки: `` и ``
- Нумерованные списки: `` и ``
- Списки определений: `<DL>` и `</DL>`
- Вложенные списки
- Форматированный текст: `<PRE>` и `</PRE>`
- Текст с отступом: `<BLOCKQUOTE>` и `</BLOCKQUOTE>`

5. За пределами HTML-файла

- Связывание
- Изображения в HTML-документе

6. Тайное становится явным, или Видимое действие невидимых меток

- Цветовая гамма HTML-документа
- Заголовок HTML-документа: что в нем может быть интересного?

7. Таблицы

- Для чего нужны таблицы?
- Как устроена таблица

8. Формы

- Для чего нужны формы?
- Как устроена форма
 - Простейшая форма
 - Как форма собирает данные
- Маленькая хитрость, или Как отправить форму почтой

9. Некоторые (предположительно полезные) советы по разработке и размещению HTML-страниц

- Правила хорошего тона при разработке HTML-документов
- Инструменты разработчика: пристрастные советы.

1. Вводные замечания

Для чего нужно это руководство?

Это руководство задумано как учебник для тех, кто хочет публиковать документацию любого рода на глобальной компьютерной сети Интернет. Насколько велика Интернет, точно не знает никто. Ходят слухи, что в 1996 году число крупных машин (серверов) и локальных сетей, подключенных к

Интернет, перевалило за 10 миллионов. Точное число пользователей просто никому не известно. Известно только, что оно очень велико – несколько сот миллионов человек по всему миру.

Чтобы опубликовать документ (здесь и далее под документом понимается файл, содержащий некоторую информацию) на Интернет, достаточно поместить его на сервер, постоянно подключенный к Интернет и способный общаться с другими серверами с помощью протокола передачи гипертекстов (HyperText Transfer Protocol, или <http://>). Совокупность таких серверов получила название "всемирной паутины" (World Wide Web, или WWW).

Разработку документов, однако, можно проводить и на компьютере, не имеющем выхода в Интернет. Автор, например, разрабатывал Руководство в текстовом процессоре *Microsoft Word 2.0*, а затем перенес в формат HTML и разместил на сервере *Yellow Internet* (<http://www.yi.com/home/ChuvakhinNikolai/>). Любой желающий может бесплатно разместить на этом сервере свои файлы суммарным объемом до двух мегабайт.

Автор приветствует распространение Руководства в печатном и электронном виде при соблюдении трех условий:

1. Руководство распространяется целиком и бесплатно.
2. Руководство не подвергается модификации без согласия автора.
3. Руководство распространяется с указанием следующих сведений об авторе: имя – Николай Чувахин, адрес электронной почты – nc@iname.com, сервер Интернет – <http://www.yi.com/home/ChuvakhinNikolai/>.

Любое отступление от указанных выше условий должно быть предварительно обсуждено с автором.

Автор будет очень рад узнать мнение читателей о качестве его работы. Очень приятно было бы получить предложения о дальнейшем совершенствовании Руководства и сотрудничестве по его распространению (включая предложения об издании в виде книги).

Что такое HTML?

Термин HTML (HyperText Markup Language) означает "язык маркировки гипертекстов". Первую версию HTML разработал сотрудник Европейской лаборатории физики элементарных частиц Тим Бернерс-Ли.

Со времени создания первой версии HTML претерпел некоторые изменения. Как и многое другое в компьютерном мире, версии, или **спецификации**, HTML оказались пронумерованными. Известны спецификации 2.0, 3.0 и 3.2. Текущую спецификацию HTML всегда можно найти на сервере W3C (<http://www.w3.org/>).

Спецификация 3.2, датированная 11 января 1997 года, находится на <http://www.w3.org/pub/WWW/TR/REC-html32.html>. Изложение этой спецификации на русском языке будет размещено (если уже не размещено) на <http://www.yi.com/home/ChuvakhinNikolai/html32.htm>.

Все, о чем написано в Руководстве, будет гарантированно работать с программным обеспечением, поддерживающим любую известную на сегодня спецификацию HTML. Естественно, для этого пришлось опустить некоторые детали, которые описаны не во всех спецификациях или описаны в разных спецификациях по-разному. Однако могу Вас уверить, что Ваши документы будут полностью работоспособными и вполне приличными по внешнему виду.

Что Вам понадобится для освоения HTML?

Для освоения HTML по этому Руководству Вам понадобятся две вещи:

1. Любой **браузер**, т.е., программа, пригодная для просмотра HTML-файлов. Мне лично очень нравится *Netscape Navigator*, поэтому именно его команды я и буду приводить, описывая операции с браузером.
2. Любой редактор текстовых файлов, поддерживающий русский язык в выбранной Вами кодировке. Если на Вашем компьютере установлен *Windows*, вполне подойдет *Notepad*.

Мы будем использовать текстовый редактор для подготовки HTML-файлов, а браузер – как инструмент контроля за сделанным.

Свои первые HTML-файлы Вы будете разрабатывать у себя на локальном диске. Другими словами, компьютер, на котором Вы будете заниматься, может и не иметь подключения к Интернет. При этом один и тот же *.htm-файл может быть одновременно открыт и в *Notepad*, и в *Netscape Navigator*. Сохранив изменения в *Notepad*, просто нажмите кнопку Reload ("перезагрузить") в *Netscape Navigator*, чтобы увидеть эти изменения реализованными в HTML-документе.

Необходимое отступление о редакторах HTML

В настоящее время широко используются два типа редакторов HTML:

1. Редакторы типа "что видишь, то и получишь" (*Netscape Navigator Gold*, *Microsoft Front Page*). Пользователь не видит "внутренностей" документа, с которым он работает, точно так же, как при работе с текстовым процессором типа *Microsoft Word* или *Word Perfect*. Кстати говоря, существует довольно много конвертеров, способных преобразовывать документы, созданные в *Microsoft Word* или *Word Perfect*, в HTML-документы.
2. Редакторы собственно HTML-текстов (*HotDog*, *Ken Nesbitt Web Editor* и многие другие). В процессе работы пользователь видит внутреннее содержание HTML-файла и может изменять его либо вручную, либо вызывая команды меню для вставки определенных элементов HTML. Работа с таким редактором очень похожа на работу с интерактивной средой программирования типа *Microsoft Visual Basic* или *Borland Delphi*.

Как я уже говорил, нам для изучения HTML понадобится только текстовый редактор и браузер. Когда Вы освоите HTML в минимальной степени, Вы сами сможете подобрать себе редактор по вкусу.

2. Наш первый HTML-документ

Как устроен HTML-документ

HTML-документ — это просто текстовый файл с расширением *.htm (Unix-системы могут содержать файлы с расширением *.html). Вот самый простой HTML-документ:

```
<html>
  <head>
    <title>
      Пример 1
    </title>
  </head>
  <body>
    <H1>
      Привет!
    </H1>
    <P>
      Это простейший пример HTML-документа.
    </P>
    <P>
      Этот *.htm-файл может быть одновременно открыт
      и в Notepad, и в Netscape. Сохранив изменения в Notepad,
      просто нажмите кнопку Reload ('перезагрузить') в Netscape,
      чтобы увидеть эти изменения реализованными в HTML-документе.
    </P>
  </body>
</html>
```

Если хотите, можно [посмотреть этот пример](#) прямо сейчас.

Для удобства чтения я ввел дополнительные отступы, однако в HTML это совсем не обязательно. Более того, браузеры просто игнорируют символы конца строки и множественные пробелы в HTML-файлах. Поэтому наш пример вполне мог бы выглядеть и вот так:

```
<html>
<head>
<title>Пример 1</title>
</head>
<body>
<H1>Привет!</H1>
<P>Это простейший пример HTML-документа.</P>
<P>Этот *.htm-файл может быть одновременно открыт и в Notepad, и в Netscape.
Сохранив изменения в Notepad, просто нажмите кнопку Reload ('перезагрузить')
в Netscape, чтобы увидеть эти изменения реализованными в HTML-документе.</P>
</body>
</html>
```

Как видно из примера, вся информация о форматировании документа сосредоточена в его фрагментах, заключенных между знаками "<" и ">". Такой фрагмент (например, <html>) называется меткой (по-английски † tag, читается "тэг").

Большинство HTML-меток † парные, то есть на каждую открывающую метку вида <tag> есть закрывающая метка вида </tag> с тем же именем, но с добавлением "/".

Метки можно вводить как большими, так и маленькими буквами. Например, метки <body>, <BODY> и <Body> будут восприняты браузером одинаково.

Многие метки, помимо имени, могут содержать **атрибуты** † элементы, дающие дополнительную информацию о том, как браузер должен обработать текущую метку. В нашем простейшем документе, однако, нет ни одного атрибута. Но мы обязательно встретимся с атрибутами уже в следующем разделе.

Обязательные метки

```
<html> ... </html>
```

Метка <html> должна открывать HTML-документ. Аналогично, метка </html> должна завершать HTML-документ.

```
<head> ... </head>
```

Эта пара меток указывает на начало и конец заголовка документа. Помимо наименования документа (см. описание метки <title> ниже), в этот раздел может включаться множество служебной информации, о которой мы обязательно поговорим чуть позже.

```
<title> ... </title>
```

Все, что находится между метками <title> и </title>, толкуется браузером как название документа. *Netscape Navigator*, например, показывает название текущего документа в заголовке окна и печатает его в левом верхнем углу каждой страницы при выводе на принтер. Рекомендуется название не длиннее 64 символов.

```
<body> ... </body>
```

Эта пара меток указывает на начало и конец тела HTML-документа, каковое тело, собственно, и определяет содержание документа.

```
<H1> ... </H1> † <H6> ... </H6>
```

Метки вида <Ni> (где i † цифра от 1 до 6) описывают заголовки шести различных уровней. Заголовок первого уровня † самый крупный, шестого уровня, естественно † самый мелкий.

```
<P> ... </P>
```

Такая пара меток описывает абзац. Все, что заключено между <P> и </P>, воспринимается как один абзац.

Метки <Ni> и <P> могут содержать дополнительный атрибут ALIGN (читается "элайн", от английского "выравнивать"), например:

```
<H1 ALIGN=CENTER>Выравнивание заголовка по центру</H1>
```

или

```
<P ALIGN=RIGHT>Образец абзаца с выравниванием по правому краю</P>
```

Подытожим все, что мы знаем, с помощью [примера 2](#):

```
<html>
```

```
<head>
```

```
<title>Пример 2</title>
```

```
</head>
```

```
<body>
```

```
<H1 ALIGN=CENTER>Привет!</H1>
```

```
<H2>Это чуть более сложный пример HTML-документа</H2>
```

```
<P>Теперь мы знаем, что абзац можно выравнивать не только влево, </P>
```

```
<P ALIGN=CENTER>но и по центру</P> <P ALIGN=RIGHT>или по правому краю.</P>
```

```
</body>
```

```
</html>
```

С этого момента Вы знаете достаточно, чтобы создавать простые HTML-документы самостоятельно от начала до конца. В следующем разделе мы поговорим, как можно улучшить наш простой HTML-документ. Начнем с малого ‡ с абзаца.

3. Внутри абзаца

Непарные метки

В этом разделе мы поговорим о метках, которые не подчиняются двум основным правилам HTML: все они непарные, а некоторые (так называемые &-последовательности) к тому же должны вводиться только маленькими буквами.

```
<BR>
```

Эта метка используется, если необходимо перейти на новую строку, не прерывая абзаца. Очень удобно при публикации стихов (см. [пример 3](#)).

```
<html>
<head>
<title>Пример 3</title>
</head>
<body>
<H1>Стих</H1>
<H2>Автор неизвестен</H2>
<P>Однажды в студеную зимнюю пору<BR>
Сижу за решеткой в темнице сырой.<BR>
Гляжу - поднимается медленно в гору<BR>
Вскормленный в неволе орел молодой.</P>
<P>И шествуя важно, в спокойствии чинном,<BR>
Мой грустный товарищ, махая крылом,<BR>
В больших сапогах, в полушубке овчинном,<BR>
Кровавую пищу клюет под окном.</P>
</body> </html>
<HR>
```

Метка <HR> описывает вот такую горизонтальную линию:

Метка может дополнительно включать атрибуты SIZE (определяет толщину линии в пикселах) и/или WIDTH (определяет размах линии в процентах от ширины экрана). В [примере 4](#) приведена небольшая коллекция горизонтальных линий.

```
<html>
<head>
<title>Пример 4</title>
</head>
<body>
<H1>Коллекция горизонтальных линий</H1>
<HR SIZE=2 WIDTH=100%><BR>
<HR SIZE=4 WIDTH=50%><BR>
<HR SIZE=8 WIDTH=25%><BR>
<HR SIZE=16 WIDTH=12%><BR>
</body>
</html>
```

&-последовательности

Поскольку символы "<" и ">" воспринимаются браузерами как начало и конец HTML-меток, возникает вопрос: а как показать эти символы на экране? В HTML это делается с помощью &-последовательностей (их еще называют символьными объектами или эскейп-последовательностями). Браузер показывает на экране символ "<", когда встречает в тексте последовательность < (по первым буквам английских слов less than ‡ меньше, чем). Знак ">" кодируется последовательностью > (по первым буквам английских слов greater than ‡ больше, чем).

Символ "&" (амперсанд) кодируется последовательностью &

Двойные кавычки (") кодируются последовательностью "

Помните: точка с запятой † обязательный элемент &-последовательности. Кроме того, все буквы, составляющие последовательность, должны быть в нижнем регистре (т.е., маленькие). Использование меток типа " или & не допускается.

Вообще говоря, &-последовательности определены для всех символов из второй половины ASCII-таблицы (куда, естественно, входят и русские буквы). Дело в том, что некоторые серверы не поддерживают восьмибитную передачу данных, и поэтому могут передавать символы с ASCII-кодами выше 127 только в виде &-последовательностей.

Комментарии

Браузеры игнорируют любой текст, помещенный между <!-- и -->. Это удобно для размещения комментариев.

<!-- Это комментарий -->

Форматирование шрифта

HTML допускает два подхода к шрифтовому выделению фрагментов текста. С одной стороны, можно прямо указать, что шрифт на некотором участке текста должен быть **жирным** или *наклонным*, то есть изменить **физический стиль** текста. С другой стороны, можно пометить некоторый фрагмент текста как имеющий некоторый отличный от нормального **логический стиль**, оставив интерпретацию этого стиля браузеру. Поясним это на примерах.

Физические стили

Под физическим стилем принято понимать прямое указание браузеру на модификацию текущего шрифта. Например, все, что находится между метками и , будет написано **жирным шрифтом**. Текст между метками <I> и </I> будет написан *наклонным шрифтом*.

Несколько особняком стоит пара меток <TT> и </TT>. Текст, размещенный между этими метками, будет написан шрифтом, имитирующим пишущую машинку, то есть имеющим фиксированную ширину символа.

Логические стили

При использовании логических стилей автор документа не может знать заранее, что увидит на экране читатель. Разные браузеры толкуют одни и те же метки логических стилей по-разному. Некоторые браузеры игнорируют некоторые метки вообще и показывают нормальный текст вместо выделенного логическим стилем. Вот самые распространенные логические стили.

** ... **

От английского emphasis † акцент.

** ... **

От английского strong emphasis † сильный акцент.

<CODE> ... </CODE>

Рекомендуется использовать для фрагментов исходных текстов.

<SAMP> ... </SAMP>

От английского sample † образец. Рекомендуется использовать для демонстрации образцов сообщений, выводимых на экран программами.

<KBD> ... </KBD>

От английского keyboard † клавиатура. Рекомендуется использовать для указания того, что нужно ввести с клавиатуры.

<VAR> ... </VAR>

От английского variable † переменная. Рекомендуется использовать для написания имен переменных.

Пример

Подытожим наши знания о логических и физических стилях с помощью [примера 5](#). Заодно Вы сможете увидеть, как Ваш браузер показывает те или иные логические стили.

```
<html>
```

```
<head>
```

```
<title>Пример 5</title>
```

```

</head>
<body>
<H1>Шрифтовое выделение фрагментов текста</H1>
<P>Теперь мы знаем, что фрагменты текста можно выделять
<B>жирным</B> или <I>наклонным</I> шрифтом. Кроме того, можно
включать в текст фрагменты с фиксированной шириной символа
<TT>(имитация пишущей машинки)</TT></P>
<P>Кроме того, существует ряд логических стилей:</P>
<P><EM>EM - от английского emphasis - акцент </EM><BR>
<STRONG>STRONG - от английского strong emphasis - сильный акцент </STRONG><BR>
<CODE>CODE - для фрагментов исходных текстов</CODE><BR>
<SAMP>SAMP - от английского sample - образец </SAMP><BR>
<KBD>KBD - от английского keyboard - клавиатура</KBD><BR>
<VAR>VAR - от английского variable - переменная </VAR></P>
</body>
</html>

```

4. Организация текста внутри документа

HTML позволяет определять внешний вид целых абзацев текста. Абзацы можно организовывать в списки, выводить их на экран в отформатированном виде, или увеличивать левое поле. Разберем все по порядку.

**Ненумерованные списки: ... **

Текст, расположенный между метками и , воспринимается как ненумерованный список. Каждый новый элемент списка следует начинать с метки . Например, чтобы создать вот такой список:

- Иван;
- Данила;
- белая кобыла

необходим вот такой HTML-текст:

```

<UL>
<LI>Иван;
<LI>Данила;
<LI>белая кобыла
</UL>

```

Обратите внимание: у метки нет парной закрывающей метки.

**Нумерованные списки: ... **

Нумерованные списки устроены точно так же, как ненумерованные, только вместо символов, выделяющих новый элемент, используются цифры. Если слегка модифицировать наш предыдущий пример:

```

<OL>
<LI>Иван;
<LI>Данила;
<LI>белая кобыла
</OL>

```

получится вот такой список:

1. Иван;
2. Данила;
3. белая кобыла

Списки определений: <DL> ... </DL>

Список определений несколько отличается от других видов списков. Вместо меток в списках определений используются метки <DT> (от английского definition term † определяемый термин) и <DD> (от английского definition definition † определение определения). Разберем это на примере. Допустим, у нас имеется следующий фрагмент HTML-текста:

```

<DL>

```

```
<DT>HTML
```

```
<DD>Термин HTML (HyperText Markup Language) означает 'язык
маркировки гипертекстов'. Первую версию HTML разработал сотрудник
Европейской лаборатории физики элементарных частиц Тим Бернерс-Ли.
```

```
<DT>HTML-документ
```

```
<DD>Текстовый файл с расширением *.htm (Unix-системы могут
содержать файлы с расширением *.html).
```

```
</DL>
```

Этот фрагмент будет выведен на экран следующим образом:

```
HTML
```

```
    Термин HTML (HyperText Markup Language) означает 'язык маркировки гипертекстов'. Первую версию HTML
    разработал сотрудник Европейской лаборатории физики элементарных частиц Тим Бернерс-Ли.
```

```
HTML-документ
```

```
    Текстовый файл с расширением *.htm (Unix-системы могут содержать файлы с расширением *.html).
```

Обратите внимание: точно так же, как метки , метки <DT> и <DD> не имеют парных закрывающих меток.

Если определяемые термины достаточно коротки, можно использовать модифицированную открывающую метку <DL COMPACT>. Например, вот такой фрагмент HTML-текста:

```
<DL COMPACT>
```

```
<DT>А
```

```
<DD>Первая буква алфавита
```

```
<DT>Б
```

```
<DD>Вторая буква алфавита
```

```
<DT>В
```

```
<DD>Третья буква алфавита
```

```
</DL>
```

будет выведен на экран примерно так:

```
А
```

```
    Первая буква алфавита
```

```
Б
```

```
    Вторая буква алфавита
```

```
В
```

```
    Третья буква алфавита
```

Вложенные списки

Элемент любого списка может содержать в себе целый список любого вида. Число уровней вложенности в принципе не ограничено, однако злоупотреблять вложенными списками все же не следует.

Вложенные списки очень удобны при подготовке разного рода планов и оглавлений.

Наши знания о списках можно вкратце свести в [пример 6](#):

```
<html>
```

```
<head>
```

```
<title>Пример 6</title>
```

```
</head>
```

```
<body>
```

```
<H1>HTML поддерживает несколько видов списков </H1>
```

```
<DL>
```

```
<DT>Ненумерованные списки
```

```
<DD>Элементы ненумерованного списка выделяются специальным
символом и отступом слева:
```

```
<UL>
```

```
<LI>Элемент 1
```

```
<LI>Элемент 2
```

```
<LI>Элемент 3
```

```
</UL>
```


<DT>Нумерованные списки

<DD>Элементы нумерованного списка выделяются отступом слева, а также нумерацией:

Элемент 1

Элемент 2

Элемент 3

<DT>Списки определений

<DD>Этот вид списков чуть сложнее, чем два предыдущих, но и выглядит более эффектно.

<P>Помните, что списки можно встраивать один в другой, но не следует закладывать слишком много уровней вложенности. </P>

<P>Обратите внимание, что внутри элемента списка может находиться несколько абзацев. Все абзацы при этом будут иметь одинаковое левое поле. </P>

</DL>

</body>

</html>

Форматированный текст: <PRE> ... </PRE>

В самом начале мы говорили о том, что браузеры игнорируют множественные пробелы и символы конца строки. Из этого правила, однако, есть исключение.

Текст, заключенный между метками <PRE> и </PRE> (от английского preformatted † предварительно форматированный), выводится браузером на экран как есть † со всеми пробелами, символами табуляции и конца строки. Это очень удобно при создании простых таблиц.

Текст с отступом: <BLOCKQUOTE> ... </BLOCKQUOTE>

Текст, заключенный между метками <BLOCKQUOTE> и </BLOCKQUOTE>, выводится браузером на экран с увеличенным левым полем.

5. За пределами HTML-файла

Связывание

Как уже упоминалось в самом начале, сокращение HTML означает "язык маркировки гипертекстов". Про маркировку мы уже поговорили достаточно. Не пора ли перейти к гипертексту?

Прежде всего, что же такое гипертекст? В отличие от обыкновенного текста, который можно читать только от начала к концу, гипертекст позволяет осуществлять мгновенный переход от одного фрагмента текста к другому. Системы помощи многих популярных программных продуктов устроены именно по гипертекстовому принципу. При нажатии левой кнопкой мыши на некоторый выделенный фрагмент текущего документа происходит переход к некоторому заранее назначенному документу или фрагменту документа.

В HTML переход от одного фрагмента текста к другому задается с помощью метки вида:

выделенный фрагмент текста

В качестве параметра [адрес перехода] может использоваться несколько типов аргументов. Самое простое † это задать имя другого HTML-документа, к которому нужно перейти. Например:

Перейти к оглавлению

Такой фрагмент HTML-текста приведет к появлению в документе выделенного фрагмента [Перейти к оглавлению](#), при нажатии на который в текущее окно будет загружен документ pr.htm.

Обратите внимание: если в адресе перехода не указан каталог, переход будет выполнен внутри текущего каталога. Если в адресе перехода не указан сервер, переход будет выполнен на текущем сервере.

Из этого следует одно очень важное практическое соображение. Если Вы подготовили к публикации некоторую группу HTML-документов, которые ссылаются друг на друга только по имени файла и находятся в одном каталоге на Вашем компьютере, вся эта группа документов будет работать точно так же, если ее поместить в любой другой каталог на любом другом

компьютере, на локальной сети или... на Интернет! Таким образом, у Вас появляется возможность разрабатывать целые коллекции документов без подключения к Интернет, и только после окончательной готовности, подтвержденной испытаниями, помещать коллекции документов на Интернет целиком.

На практике, однако, часто бывает необходимо дать ссылку на документ, находящийся на другом сервере. Например, если Вы хотите дать ссылку на это руководство со своей странички, Вам придется ввести в свой HTML-документ примерно такой фрагмент:

```
<A HREF="http://www.yi.com/home/ChuvakhinNikolai/pr.htm">
```

```
Практическое руководство по HTML Николая Чувахина</A>
```

При необходимости можно задать переход не просто к некоторому документу, но и к определенному месту внутри этого документа. Для этого необходимо создать в документе, к которому будет задан переход, некоторую опорную точку, или **анкер**. Разберем это на примере.

Допустим, что необходимо осуществить переход из файла 1.htm к словам "Переход закончен" в файле 2.htm (файлы находятся в одном каталоге). Прежде всего, необходимо создать вот такой анкер в файле 2.htm:

```
<A NAME="AAA">Переход закончен</A>
```

Слова "Переход закончен" при этом никак не будут выделены в тексте документа.

Затем в файле 1.htm (или в любом другом) можно определить переход на этот анкер:

```
<A HREF="2.htm#AAA">Переход к анкеру AAA</A>
```

Кстати говоря, переход к этому анкеру можно определить и внутри самого документа 2.htm † достаточно только включить в него вот такой фрагмент:

```
<A HREF="#AAA">Переход к анкеру AAA</A>
```

На практике это очень удобно при создании больших документов. В начале документа можно поместить оглавление, состоящее из ссылок на анкеры, расположенные в заголовках разделов документа.

Во избежание недоразумений рекомендуется задавать имена анкеров латинскими буквами. Следите за написанием имен анкеров: большинство браузеров отличают большие буквы от маленьких. Если имя анкера определено как AAA, ссылка на анкер aaa или AaA не выведет Вас на анкер AAA, хотя документ, скорее всего, будет загружен корректно.

Пока что мы обсуждали только ссылки на HTML-документы. Однако возможны ссылки и на другие виды ресурсов:

```
<A HREF="ftp://server/directory/file.ext">Выгрузить файл</A>
```

Такая ссылка, если ей воспользоваться, запустит протокол передачи файлов и начнет выгрузку файла file.ext, находящегося в каталоге directory на сервере server, на локальный диск пользователя.

```
<A HREF="mailto:user@mail.box">Послать письмо</A>
```

Если пользователь совершит переход по такой ссылке, у него на экране откроется окно ввода исходящего сообщения его почтовой программы. В строке To: ("Куда") окна почтовой программы будет указано user@mail.box.

Разберем все, что мы знаем о связывании, с помощью [примера 7](#).

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Пример 7</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<H1>Связывание </H1>
```

```
<P>С помощью ссылок можно переходить к другим файлам (например, к <A HREF="pr.htm">оглавлению этого руководства</A>).</P>
```

```
<P>Можно выгружать файлы (например,
```

```
<A HREF="ftp://yi.com/home/ChuvakhinNikolai/html-pr.doc">это руководство в формате Microsoft Word 2.0</A>) по FTP.</P>
```

```
<P>Можно дать пользователю возможность послать почту (например,
```

```
<A HREF="mailto:nc@iname.com">автору этого руководства</A>.</P>
</BODY>
</HTML>
```

Изображения в HTML-документе

Встроить изображение в HTML-документ очень просто. Для этого нужно только иметь это самое изображение в формате GIF (файл с расширением *.gif) или JPEG (файл с расширением *.jpg или *.jpeg) и одну строчку в HTML-тексте.

Допустим, нам нужно включить в документ изображение, записанное в файл picture.gif, находящийся в одном каталоге с HTML-документом. Тогда строчка будет вот такая:

```
<IMG SRC="picture.gif">
```

Метка может также включать атрибут ALT="[текст]", например:

```
<IMG SRC="picture.gif" ALT="Картинка">
```

Встретив такую метку, браузер покажет на экране текст Картинка и начнет загружать на его место картинку из файла picture.gif. Атрибут ALT может оказаться необходимым для старых браузеров, которые не поддерживают изображений, а также на случай, если у браузера отключена автоматическая загрузка изображений (при медленном подключении к Интернет это делается для экономии времени).

Файл, содержащий изображение, может находиться в другом каталоге или даже на другом сервере.

В этом случае стоит указать его полное имя (см. раздел "[Связывание](#)").

Разберем все, что мы знаем об изображениях, с помощью [примера 8](#).

```
<HTML>
<HEAD>
<TITLE>Пример 8</TITLE>
</HEAD>
<BODY>
<H1>Изображения </H1>
<P>Изображение можно встроить очень просто: </P>
<P><IMG SRC="picture.gif"></P>
<P>Кроме того, изображение можно сделать "горячим", то есть
осуществлять переход при нажатии на изображение:</P>
<P><A HREF="pr.htm"><IMG SRC="picture.gif"></A></P>
</BODY>
</HTML>
```

Обратите внимание на вторую часть примера. Если ссылка на изображение находится между метками и , изображение фактически становится кнопкой, при нажатии на которую происходит переход по ссылке (в примере 8 переход происходит на оглавление Руководства).

6. Тайное становится явным, или Видимое действие невидимых меток

Цветовая гамма HTML-документа

Если Вы читаете это Руководство на Интернет, Вы уже могли заметить, что главы Руководства и примеры несколько отличаются по цветовой гамме. Главы Руководства написаны черным по белому, а гипертекстовые ссылки выделены красным цветом. Я не знаю, как Вы видите примеры, но у меня они написаны черным по серому с выделением ссылок синим цветом.

Дело в том, что главы Руководства содержат HTML-метки, определяющие их цветовую гамму. Примеры таких меток не содержат, поэтому их цветовая гамма устанавливается браузером самостоятельно. Именно поэтому я не знаю, какие цвета Вы видите в примерах † это определяется настройкой Вашего браузера.

Цветовая гамма HTML-документа определяется атрибутами, размещенными внутри метки <BODY>.

Вот список этих атрибутов:

bgcolor

Определяет цвет фона документа.

text

Определяет цвет текста документа.

link

Определяет цвет выделенного элемента текста, при нажатии на который происходит переход по гипертекстовой ссылке.

vlink

Определяет цвет ссылки на документ, который уже был просмотрен ранее.

alink

Определяет цвет ссылки в момент, когда на нее указывает курсор мыши и нажата ее правая кнопка, то есть непосредственно перед переходом по ссылке.

Цвет кодируется последовательностью из трех пар символов. Каждая пара представляет собой шестнадцатиричное значение насыщенности заданного цвета одним из трех основных цветов (красным, зеленым и синим) в диапазоне от нуля (00) до 255 (FF). Разберем несколько примеров.

bgcolor=#FFFFFF

Цвет фона. Насыщенность красным, зеленым и синим одинакова \ddagger FF (это шестнадцатиричное представление числа 255). Результат \ddagger белый цвет.

text=#000000

Цвет текста. Насыщенность красным, зеленым и синим одинакова \ddagger 00 (ноль). Результат \ddagger черный цвет.

link=#FF0000

Цвет гипертекстовой ссылки. Насыщенность красным \ddagger FF (255), зеленым и синим \ddagger 00 (ноль). Результат \ddagger красный цвет.

Кроме того, метка <BODY> может включать атрибут background="[имя файла]", который задает изображение, служащее фоном для текста и других изображений. Как и любое другое изображение, фон должен быть представлен в формате GIF (файл с расширением *.gif) или JPEG (файл с расширением *.jpg или *.jpeg).

Браузеры заполняют множественными копиями изображения-фона все пространство окна, в котором открыт документ, подобно тому, как при строительстве большие пространства стен покрывают маленькими (и одинаковыми) плитками.

Важно отметить, что цвет фона и изображение-фон никак не отображаются на бумаге при выводе HTML-документа на печать. Из этого есть одно важное практическое следствие: **старайтесь не использовать текст белого цвета.**

Заголовок HTML-документа: что в нем может быть интересно?

Заголовок HTML-документа, вообще говоря, не виден пользователю при просмотре. Однако в нем есть некоторые интересные особенности, которые стоит знать.

Заголовок HTML-документа может включать неограниченное количество так называемых META-инструкций. META-инструкция \ddagger это просто способ определить некоторую переменную путем указания ее имени (атрибут NAME) и значения (атрибут CONTENT). Вот некоторые наиболее типичные META-инструкции:

```
<META NAME="description" CONTENT="Это руководство - учебник для тех, кто хочет публиковать документацию любого рода на глобальной компьютерной сети Интернет">
```

Такая META-инструкция определяет переменную description, содержащую краткое описание документа. Многие поисковые механизмы постоянно сканируют Интернет в поисках HTML-файлов, отыскивают в них эту переменную, сохраняют ее в своих базах данных и демонстрируют ее в ответ на запросы пользователей.

```
<META NAME="keywords" CONTENT="Интернет, HTML, WWW, руководство, публикация, гипертекст">
```

Такая META-инструкция определяет переменную keywords, содержащую набор ключевых слов, описывающих содержание документа. На практике поиск по ключевым словам очень удобно использовать при строительстве поискового механизма, работающего в пределах одного сервера (со стороны очень похоже, что именно такой подход был использован, например, при создании сервера технической поддержки фирмы *Novell*, ведущего производителя операционных систем для локальных компьютерных сетей).

Другая группа META-инструкций определяет эквиваленты команд протокола передачи гипертекстов. (Не переживайте, это не так страшно, как кажется на первый взгляд!) Разберем несколько примеров:

```
<META HTTP-EQUIV="Content-type"
```

CONTENT="text/html; charset=windows-1251">

Эта META-инструкция дает браузеру указание интерпретировать загружаемый документ как содержащий HTML-текст в кодировке *Windows/1251*.

<META HTTP-EQUIV="Content-type"

CONTENT="text/html; charset=koi8-r">

Эта META-инструкция абсолютно аналогична предыдущей, только в качестве кодировки указана КОИ-8.

<META HTTP-EQUIV="Refresh" CONTENT="[время]; URL=[документ]">

Такая META-инструкция дает браузеру примерно такое указание: "Если через [время] секунд после завершения загрузки этого документа пользователь не перейдет к другому документу, начать загрузку ресурса [документ]". Более конкретно это может выглядеть, к примеру, вот так:

<META HTTP-EQUIV="Refresh"

CONTENT="10; URL=http://www.yi.com/home/ChuvakhinNikolai/">

Если пользователь не предпримет никаких видимых действий в течение 10 секунд после загрузки документа, содержащего такую инструкцию, автоматически будет загружен документ <http://www.yi.com/home/ChuvakhinNikolai/>.

META-инструкцию Refresh можно использовать, например, если Вы перенесли некоторый документ с одного сервера на другой. Вместо копии документа на старом сервере можно оставить короткое сообщение о переносе, включающее META-инструкцию Refresh и адрес документа на новом сервере.

Если в качестве параметра [документ] подставить имя файла, содержащего звук, через [время] секунд после загрузки HTML-файла начнется загрузка и проигрывание этого звука (при условии, конечно, что браузер пользователя поддерживает формат этого звукового файла). Очень удобно для всякого рода приветственных речей.

Пример

В отличие от всех примеров, которые мы рассматривали ранее, [пример 9](#) состоит не из одного, а из трех файлов. Используя META-инструкцию Refresh, мы создадим небольшой слайд-фильм из трех кадров (файлов e0009.htm, e0009a.htm и e0009b.htm), которые будут циклически повторяться. Для остановки демонстрации нужно будет воспользоваться любой из гипертекстовых ссылок.

```
<HTML><!--файл e0009.htm -->
```

```
<HEAD>
```

```
<TITLE>Пример 9</TITLE>
```

```
<META HTTP-EQUIV="Refresh" CONTENT="2; URL=e0009a.htm">
```

```
</HEAD>
```

```
<BODY bgcolor=#FFFFFF text=#000000 link=#FF0000>
```

```
<H1>Слайд-демонстрация цветовых гамм <BR>
```

```
с помощью META-инструкции Refresh </H1>
```

```
<P>Черный текст на белом фоне </P>
```

```
<P><A HREF="pr.htm">Возврат к оглавлению</A>|
```

```
<A HREF="pr0006.htm">Возврат к главе 6</A></P>
```

```
</BODY>
```

```
</HTML><!--конец файла e0009.htm -->
```

```
<HTML><!--файл e0009a.htm -->
```

```
<HEAD>
```

```
<TITLE>Пример 9a</TITLE>
```

```
<META HTTP-EQUIV="Refresh" CONTENT="2; URL=e0009b.htm">
```

```
</HEAD>
```

```
<BODY bgcolor=#000000 text=#FFFFFF link=#FF0000>
```

```
<H1>Слайд-демонстрация цветовых гамм <BR>
```

```
с помощью META-инструкции Refresh </H1>
```

```
<P>Белый текст на черном фоне </P>
```

```
<P><A HREF="pr.htm">Возврат к оглавлению</A>|
```

```
<A HREF="pr0006.htm">Возврат к главе 6</A></P>
```

```

</BODY>
</HTML><!--конец файла e0009a.htm -->
<HTML><!--файл e0009b.htm -->
<HEAD>
<TITLE>Пример 9b</TITLE>
<META HTTP-EQUIV="Refresh" CONTENT="2; URL=e0009.htm">
</HEAD>
<BODY bgcolor=#C0C0C0 text=#0000FF link=#FF0000>
<H1>Слайд-демонстрация цветовых гамм <BR>
с помощью META-инструкции Refresh </H1>
<P>Синий текст на сером фоне </P>
<P>[<A HREF="pr.htm">Возврат к оглавлению</A>|
<A HREF="pr0006.htm">Возврат к главе 6</A>]</P>
</BODY>
</HTML><!--конец файла e0009b.htm -->

```

7. Таблицы

Для чего нужны таблицы?

На этот вопрос есть очевидный ответ: таблицы нужны для представления информации в табличном виде. Есть, однако, и менее очевидные ответы.

До настоящего времени мы имели дело с документами, в которых существовал только один "поток" текста. На практике иногда очень хочется расположить текст в несколько колонок. Таблица может в этом помочь.

Кроме того, таблица, состоящая из одной ячейки, может очень эффектно выделить фрагмент текста, на который Вы хотите обратить внимание читателя.

Как устроена таблица

В устройстве таблицы легче всего разобраться на простом [примере](#).

```

<HTML>
<HEAD>
<TITLE>Пример 10</TITLE>
</HEAD>
<H1>Простейшая таблица </H1>
<TABLE BORDER=1> <!--Это начало таблицы-->
<CAPTION> <!--Это заголовок таблицы-->
У таблицы может быть заголовок
</CAPTION>
<TR> <!--Это начало первой строки-->
<TD> <!--Это начало первой ячейки-->
Первая строка, первая колонка
</TD> <!--Это конец первой ячейки-->
<TD> <!--Это начало второй ячейки-->
Первая строка, вторая колонка
</TD> <!--Это конец второй ячейки-->
</TR> <!--Это конец первой строки-->
<TR> <!--Это начало второй строки-->
<TD> <!--Это начало первой ячейки-->
Вторая строка, первая колонка
</TD> <!--Это конец первой ячейки-->
<TD> <!--Это начало второй ячейки-->
Вторая строка, вторая колонка
</TD> <!--Это конец второй ячейки-->
</TR> <!--Это конец второй строки-->
</TABLE> <!--Это конец таблицы-->

```

</BODY>

</HTML>

Таблица начинается с метки <TABLE> и заканчивается меткой </TABLE>. Метка <TABLE> может включать несколько атрибутов:

ALIGN

Устанавливает расположение таблицы по отношению к полям документа. Допустимые значения: ALIGN=LEFT (выравнивание влево), ALIGN=CENTER (выравнивание по центру), ALIGN=RIGHT (выравнивание вправо).

WIDTH

Ширина таблицы. Ее можно задать в пикселах (например, WIDTH=400) или в процентах от ширины страницы (например, WIDTH=80%).

BORDER

Устанавливает ширину внешней рамки таблицы и ячеек в пикселах (например, BORDER=4). Если атрибут не установлен, таблица показывается без рамки.

CELLSPACING

Устанавливает расстояние между рамками ячеек таблицы в пикселах (например, CELLSPACING=2).

CELLPADDING

Устанавливает расстояние между рамкой ячейки и текстом в пикселах (например, CELLPADDING=10).

Таблица может иметь заголовок (<CAPTION> ... </CAPTION>), хотя заголовок не является обязательным. Метка <CAPTION> может включать атрибут ALIGN. Допустимые значения: <CAPTION ALIGN=TOP> (заголовок помещается над таблицей) и <CAPTION ALIGN=BOTTOM> (заголовок помещается под таблицей).

Каждая строка таблицы начинается с метки <TR> и заканчивается меткой </TR>. Метка <TR> может включать следующие атрибуты:

ALIGN

Устанавливает выравнивание текста в ячейках строки. Допустимые значения: ALIGN=LEFT (выравнивание влево), ALIGN=CENTER (выравнивание по центру), ALIGN=RIGHT (выравнивание вправо).

VALIGN

Устанавливает вертикальное выравнивание текста в ячейках строки. Допустимые значения: VALIGN=TOP (выравнивание по верхнему краю), VALIGN=MIDDLE (выравнивание по центру), VALIGN=BOTTOM (выравнивание по нижнему краю).

Каждая ячейка таблицы начинается с метки <TD> и заканчивается меткой </TD>. Метка <TD> может включать следующие атрибуты:

NOWRAP

Присутствие этого атрибута означает, что содержимое ячейки должно быть показано в одну строку.

COLSPAN

Устанавливает "размах" ячейки по горизонтали. Например, COLSPAN=3 означает, что ячейка простирается на три колонки.

ROWSPAN

Устанавливает "размах" ячейки по вертикали. Например, ROWSPAN=2 означает, что ячейка занимает две строки.

ALIGN

Устанавливает выравнивание текста в ячейке. Допустимые значения: ALIGN=LEFT (выравнивание влево), ALIGN=CENTER (выравнивание по центру), ALIGN=RIGHT (выравнивание вправо).

VALIGN

Устанавливает вертикальное выравнивание текста в ячейке. Допустимые значения: VALIGN=TOP (выравнивание по верхнему краю), VALIGN=MIDDLE (выравнивание по центру), VALIGN=BOTTOM (выравнивание по нижнему краю).

WIDTH

Устанавливает ширину ячейки в пикселах (например, WIDTH=200).

HEIGHT

Устанавливает высоту ячейки в пикселах (например, HEIGHT=40).

Если ячейка таблицы пуста, вокруг нее не рисуется рамка. Если ячейка пуста, а рамка нужна, в ячейку можно ввести символьный объект ` ` (non-breaking space `‡` неразрывающий пробел). Ячейка по-прежнему будет пустой, а рамка вокруг нее будет.

Любопытно отметить, что любая ячейка таблицы может содержать в себе другую таблицу.

8. Формы

Для чего нужны формы?

Форма `‡` это инструмент, с помощью которого HTML-документ может послать некоторую информацию в некоторую заранее определенную точку внешнего мира, где информация будет некоторым образом обработана.

Рассказать о формах в книге, посвященной HTML, достаточно трудно. Причина очень простая: создать форму гораздо проще, чем ту "точку внешнего мира", в которую форма будет посылать информацию. В качестве такой "точки" в большинстве случаев выступает программа, написанная на Перл или Си. Программы, обрабатывающие данные, переданные формами, часто называют CGI-скриптами. Сокращение CGI (Common Gateways Interface) означает "общепринятый интерфейс шлюзов". Написание CGI-скриптов в большинстве случаев требует хорошего знания соответствующего языка программирования и возможностей операционной системы Unix.

В последнее время определенное распространение получил язык PHP/PI, инструкции которого можно встраивать прямо в HTML-документы (документы при этом сохраняются в виде файлов с расширением *.pht или *.php).

Формы передают информацию программам-обработчикам в виде пар [имя переменной]=[значение переменной]. Имена переменных следует задавать латинскими буквами. Значения переменных воспринимаются обработчиками как строки, даже если они содержат только цифры.

Как устроена форма

Форма открывается меткой `<FORM>` и заканчивается меткой `</FORM>`. HTML-документ может содержать в себе несколько форм, однако формы не должны находиться одна внутри другой. HTML-текст, включая метки, может размещаться внутри форм без ограничений.

Метка `<FORM>` может содержать три атрибута, один из которых является обязательным. Вот эти атрибуты:

ACTION

Обязательный атрибут. Определяет, где находится обработчик формы.

METHOD

Определяет, каким образом (иначе говоря, с помощью какого метода протокола передачи гипертекстов) данные из формы будут переданы обработчику. Допустимые значения: `METHOD=POST` и `METHOD=GET`. Если значение атрибута не установлено, по умолчанию предполагается `METHOD=GET`.

ENCTYPE

Определяет, каким образом данные из формы будут закодированы для передачи обработчику. Если значение атрибута не установлено, по умолчанию предполагается `ENCTYPE=application/x-www-form-urlencoded`.

Простейшая форма

Для того, чтобы запустить процесс передачи данных из формы обработчику, нужен какой-то орган управления. Создать такой орган управления очень просто:

```
<INPUT TYPE=submit>
```

Встретив такую строчку внутри формы, браузер нарисует на экране кнопку с надписью Submit (читается "сабмит" с ударением на втором слоге, от английского "подавать"), при нажатии на которую все имеющиеся в форме данные будут переданы обработчику, определенному в метке `<FORM>`.

Надпись на кнопке можно задать такую, какая нравится, путем введения атрибута `VALUE="[Надпись]"` (читается "взлю" с ударением на первом слоге, от английского "значение"), например:

```
<INPUT TYPE=submit VALUE="Поехали!">
```

Теперь мы знаем достаточно для того, чтобы написать простейшую форму ([пример 11](#)). Она не будет собирать никаких данных, а просто вернет нас к тексту этой главы.

```
<HTML>
```

```
<HEAD>
```



```

<TITLE>Пример 11</TITLE>
</HEAD>
<H1>Простейшая форма </H1>
<FORM ACTION="pr0008.htm"> <!--Это начало формы-->
<INPUT TYPE=submit VALUE="Назад, к главе 8...">
</FORM> <!--Это конец формы-->
</BODY>
</HTML>

```

Надпись, нанесенную на кнопку, можно при необходимости передать обработчику путем введения в определение кнопки атрибута NAME=[имя] (читается "нэйм", от английского "имя"), например:

```
<INPUT TYPE=submit NAME=button VALUE="Поехали!">
```

При нажатии на такую кнопку обработчик вместе со всеми остальными данными получит и переменную button со значением Поехали!.

В форме может быть несколько кнопок типа submit с различными именами и/или значениями. Обработчик, таким образом, может действовать по-разному в зависимости от того, какую именно кнопку submit нажал пользователь.

Как форма собирает данные

Существуют и другие типы элементов <INPUT>. Каждый элемент <INPUT> должен включать атрибут NAME=[имя], определяющий имя элемента (и, соответственно, имя переменной, которая будет передана обработчику). Имя должно задаваться **только латинскими буквами**. Большинство элементов <INPUT> должны включать атрибут VALUE="[значение]", определяющий значение, которое будет передано обработчику под этим именем. Для элементов <INPUT TYPE=text> и <INPUT TYPE=password>, однако, этот атрибут не обязателен, поскольку значение соответствующей переменной может вводиться пользователем с клавиатуры.

Основные типы элементов <INPUT>:

TYPE=text

Определяет окно для ввода строки текста. Может содержать дополнительные атрибуты SIZE=[число] (ширина окна ввода в символах) и MAXLENGTH=[число] (максимально допустимая длина вводимой строки в символах). Пример:

```
<INPUT TYPE=text SIZE=20 NAME=user VALUE="Иван">
```

Определяет окно шириной 20 символов для ввода текста. По умолчанию в окне находится текст Иван, который пользователь может редактировать. Отредактированный (или неотредактированный) текст передается обработчику в переменной user.

TYPE=password

Определяет окно для ввода пароля. Абсолютно аналогичен типу text, только вместо символов вводимого текста показывает на экране звездочки (*). Пример:

```
<INPUT TYPE=password NAME=pw SIZE=20 MAXLENGTH=10>
```

Определяет окно шириной 20 символов для ввода пароля. Максимально допустимая длина пароля ± 10 символов. Введенный пароль передается обработчику в переменной pw.

TYPE=radio

Определяет радиокнопку. Может содержать дополнительный атрибут checked (показывает, что кнопка помечена). В группе радиокнопок с одинаковыми именами может быть только одна помеченная радиокнопка. Пример:

```

<INPUT TYPE=radio NAME=modem VALUE="9600" checked> 9600 бит/с
<INPUT TYPE=radio NAME=modem VALUE="14400"> 14400 бит/с
<INPUT TYPE=radio NAME=modem VALUE="28800"> 28800 бит/с

```

Определяет группу из трех радиокнопок, подписанных 9600 бит/с, 14400 бит/с и 28800 бит/с. Первоначально помечена первая из кнопок. Если пользователь не отметит другую кнопку, обработчику будет передана переменная modem со значением 9600. Если пользователь отметит другую кнопку, обработчику будет передана переменная modem со значением 14400 или 28800.

TYPE=checkbox

Определяет квадрат, в котором можно сделать пометку. Может содержать дополнительный атрибут checked (показывает, что квадрат помечен). В отличие от радиокнопок, в группе квадратов с одинаковыми именами может быть несколько помеченных квадратов. Пример:

```
<INPUT TYPE=checkbox NAME=comp VALUE="PC"> Персональные компьютеры
<INPUT TYPE=checkbox NAME=comp VALUE="WS" checked> Рабочие станции
<INPUT TYPE=checkbox NAME=comp VALUE="LAN"> Серверы локальных сетей
<INPUT TYPE=checkbox NAME=comp VALUE="IS" checked> Серверы Интернет
```

Определяет группу из четырех квадратов. Первоначально помечены второй и четвертый квадраты. Если пользователь не произведет изменений, обработчику будут переданы две переменные: comp=WS и comp=IS.

TYPE=hidden

Определяет скрытый элемент данных, который не виден пользователю при заполнении формы и передается обработчику без изменений. Такой элемент иногда полезно иметь в форме, которая время от времени подвергается переработке, чтобы обработчик мог знать, с какой версией формы он имеет дело. Другие возможные варианты использования Вы вполне можете придумать сами. Пример:

```
<INPUT TYPE=hidden NAME=version VALUE="1.1">
```

Определяет скрытую переменную version, которая передается обработчику со значением 1.1.

TYPE=reset

Определяет кнопку, при нажатии на которую форма возвращается в исходное состояние. Поскольку при использовании этой кнопки данные обработчику не передаются, кнопка типа reset может и не иметь атрибута name. Пример:

```
<INPUT TYPE=reset VALUE="Очистить поля формы">
```

Определяет кнопку Очистить поля формы, при нажатии на которую форма возвращается в исходное состояние.

Помимо элементов <INPUT>, формы могут содержать меню <SELECT> и поля для ввода текста <TEXTAREA>.

Меню <SELECT> из n элементов выглядит примерно так:

```
<SELECT NAME="[имя]">
<OPTION VALUE="[значение 1]">[текст 1]
<OPTION VALUE="[значение 2]">[текст 2]
...
<OPTION VALUE="[значение n]">[текст n]
</SELECT>
```

Как Вы видите, меню начинается с метки <SELECT> и заканчивается меткой </SELECT>. Метка <SELECT> содержит обязательный атрибут NAME, определяющий имя переменной, которую генерирует меню.

Метка <SELECT> может также содержать атрибут MULTIPLE, присутствие которого показывает, что из меню можно выбрать несколько элементов. Большинство браузеров показывают меню <SELECT MULTIPLE> в виде окна, в котором находятся элементы меню (высоту окна в строках можно задать атрибутом SIZE=[число]). Меню <SELECT> в большинстве случаев показывается в виде выпадающего меню.

Метка <OPTION> определяет элемент меню. Обязательный атрибут VALUE устанавливает значение, которое будет передано обработчику, если выбран этот элемент меню. Метка <OPTION> может включать атрибут checked, показывающий, что данный элемент отмечен по умолчанию.

Разберем небольшой пример.

```
<SELECT NAME="selection">
<OPTION VALUE="option1" checked>Вариант 1
<OPTION VALUE="option2">Вариант 2
<OPTION VALUE="option3">Вариант 3
</SELECT>
```

Такой фрагмент определяет меню из трех элементов: Вариант 1, Вариант 2 и Вариант 3. По умолчанию выбран элемент Вариант 1. Обработчику будет передана переменная selection значение которой может быть option1 (по умолчанию), option2 или option3.

После всего, что мы уже узнали, элемент <TEXTAREA> может показаться совсем простым. Например:

```
<TEXTAREA NAME=address ROWS=5 COLS=50>
```

А здесь - Ваш адрес...

</TEXTAREA>

Все атрибуты обязательны. Атрибут NAME определяет имя, под которым содержимое окна будет передано обработчику (в примере ‡ address). Атрибут ROWS устанавливает высоту окна в строках (в примере ‡ 5). Атрибут COLS устанавливает ширину окна в символах (в примере ‡ 50).

Текст, размещенный между метками <TEXTAREA> и </TEXTAREA>, представляет собой содержимое окна по умолчанию. Пользователь может его отредактировать или просто стереть.

Важно знать, что русские буквы в окне <TEXTAREA> при передаче обработчику могут быть конвертированы в соответствующие им символьные объекты.

Пример

Для демонстрации использования форм я написал небольшую программу на PHP, которая находится по адресу:

<http://206.31.82.215/hp/nc/fd-win.pht>

Исходные данные в эту программу передаст форма, описанная в [примере 12](#):

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Пример 12</TITLE>
```

```
</HEAD>
```

```
<H1>Несколько более сложная форма </H1>
```

```
<FORM ACTION="http://206.31.82.215/hp/nc/fd-win.pht" METHOD=post>
```

```
<H2>Расскажите немного о себе...</H2>
```

```
<P>Указывать подлинные данные совсем не обязательно.
```

```
Для целей демонстрации вполне подойдут и вымышленные. </P>
```

```
<P>Имя: <INPUT TYPE=text SIZE=40 NAME=fn><BR>
```

```
Фамилия: <INPUT TYPE=text SIZE=40 NAME=ln><BR>
```

```
Пол: <INPUT TYPE=radio NAME=gender VALUE="male" checked>мужской
```

```
<INPUT TYPE=radio NAME=gender VALUE="female">женский<BR>
```

```
Возраст: <INPUT TYPE=text SIZE=5 NAME=age> лет<BR>
```

```
<INPUT TYPE=submit VALUE="Запустить обработчик"></P>
```

```
</FORM>
```

```
</BODY>
```

```
</HTML>
```

Заполняйте форму, жмите на кнопку и смотрите, что будет...

Маленькая

хитрость,

или Как отправить форму почтой

Все это прекрасно, скажут скептики, но на кой ляд нужны формы людям, которым их нечем обработать? Отчасти это верно, но только отчасти.

HTML предоставляет в Ваше распоряжение довольно мощный механизм пересылки содержимого форм по электронной почте. Вот как это выглядит на практике.

Допустим, что мы слегка изменили Пример 12. Вместо строки

```
<FORM ACTION="http://206.31.82.215/hp/nc/fd-win.pht" METHOD=post>
```

мы ввели строку

```
<FORM ACTION="mailto:user@mail.box" ENCTYPE=text/plain>
```

Обратите внимание, что мы изменили алгоритм кодирования на text/plain, то есть фактически выключили кодирование вообще.

Предположим теперь, что пользователь указал, имя Иван, фамилию Петров, мужской пол и возраст 22 года. Теперь вопрос: что произойдет, если пользователь нажмет на кнопку Запустить обработчик?

Ответ прост. На адрес user@mail.box электронной почтой автоматически будет отправлено сообщение следующего содержания:

```
fn=Иван
```

```
ln=Петров
```

```
gender=male
```

```
age=22
```

К сожалению, не все пользователи смогут воспользоваться такой формой. Дело в том, что всю работу по составлению сообщения и запуску почтовой программы для его отправки фактически берет на себя браузер пользователя. Это значит, что конфигурация доступа пользователя к Интернет должна обеспечивать одновременное функционирование протокола передачи гипертекстов и протокола доставки исходящей почты. Такое возможно не всегда. Тем не менее, даже если это невозможно, ничего смертельного не случится. Браузер просто выдаст сообщение об ошибке.

9. Некоторые (предположительно полезные) советы по разработке и размещению HTML-страниц

Правила хорошего тона при разработке HTML-документов

Ознакомьтесь с текущей спецификацией HTML

Производители браузеров часто добавляют к своим творениям поддержку меток, не входящих в текущую спецификацию HTML. Именно поэтому иногда можно встретить страницы, оптимизированные для того или иного браузера. На мой взгляд, это далеко не лучший подход. Однако, как говорил Рэй Брэдбери, единственный способ заставить писателя не писать † это кольт сорок пятого калибра. И писатели пишут...

Не питайте абсолютного доверия к визуальным редакторам HTML

Помните, что большинство визуальных редакторов HTML, хотя и работают в целом корректно, имеют мелкие (а иногда и крупные) недостатки. Отчасти это связано с тем, что HTML с самого начала не был приспособлен для автоматизированной разработки.

Используйте продуманную иерархию заголовков

Помните, что Ваш документ будет читаться гораздо лучше, если в нем будет четкое разбиение на разделы и подразделы. HTML предоставляет в Ваше распоряжение шесть уровней заголовков. Воспользуйтесь этим богатством в своих интересах † создайте легко читаемый документ с интуитивно ясной структурой.

Следите за вложенностью меток

Современные браузеры способны правильно обработать вложенные метки. Например, вот такой фрагмент HTML-текста

```
<B><I>жирный наклонный шрифт</I></B>
```

большинство браузеров вполне успешно воспроизведут как ***жирный наклонный шрифт***.

Обратите внимание † метки <I>...</I> в этом фрагменте находятся внутри меток Другими словами, метки вложены одна в другую.

Работа браузера, однако, окажется сильно затрудненной, если вложенность окажется нарушенной, например:

```
<B><I>вложенность меток нарушена</B></I>
```

Соблюдение вложенности † очень важная часть общей культуры написания HTML-текста.

Указывайте альтернативный текст при включении в документ изображений

Помните, что пользователи могут отключать автоматическую загрузку изображений. Не ленитесь подсказать пользователю, чего именно он не видит. Это особенно важно, когда изображения являются гипертекстовыми ссылками.

При разработке крупных документов создавайте оглавления

При написании крупного документа очень полезно поместить в его начале оглавление с гипертекстовыми ссылками на разделы. Это может облегчить пользователю работу с документом, особенно если документ носит справочный характер.

Хорошо продумайте навигацию

Попытайтесь вообразить себе, что в распоряжении пользователя нет кнопки Back. Подумайте о том, как обеспечить пользователю необходимую свободу перемещения в этом случае. Это поможет Вам спроектировать удобную в пользовании коллекцию HTML-документов.

Инструменты разработчика: пристрастные советы

Предупреждаю Вас честно: все приведенные ниже комментарии представляют собой мое личное (и, вполне вероятно, необъективное) мнение.

Визуальные редакторы HTML

Netscape Navigator Gold

На мой взгляд, это лучший из редакторов типа "что видишь, то и получишь". Не сочиняет нестандартных меток, хотя некоторые стандартные (в частности, метки списков) пишет не лучшим образом. При попытке перейти из режима просмотра в режим редактирования иногда подвисает на томительные 30-40 секунд. На мой взгляд, недостаточно защищен от ошибок неопытного пользователя (хотя это относится только к подсистеме редактора; в режиме браузера работает четко).

Распространяется бесплатно. Необходимую версию можно подобрать прямо отсюда:

Программное обеспечение:

Select a Product ▼

Операционная система:

Select Your Operating System ▼

Найти, где лежит

Microsoft Front Page

Ходит по России в основном в виде версий для 30-дневных тестов с обезвреженным механизмом самоуничтожения. Наиболее крупных болячек две:

1. Вставляет в заголовок HTML-документа META-инструкцию
`<META HTTP-EQUIV="Content-type" CONTENT="text/html; charset=windows-1251">`
 Это превращается в неумную шутку при попытке опубликовать документ в кодировке КОИ-8.
 Лекарство только одно: перед размещением на сервере удалить или исправить META-инструкцию вручную.
2. Рассыпает ровным слоем по документу метки вида
`[текст]`
 для указания того, каким шрифтом показывать [текст].
 В HTML 3.2 метка `` используется для изменения размеров и цвета шрифта. Метка `` в спецификации упомянута как поддерживаемая некоторыми браузерами, но нежелательная и потому не включенная в спецификацию и не рекомендуемая к применению. В самом деле, формат HTML не привязан даже к конкретной аппаратной платформе, зачем же привязывать его к шрифту!

Небольшое отступление о не упомянутых здесь редакторах

Многие редакторы типа "что видишь, то и получишь" сохраняют русские буквы в виде соответствующих им &-последовательностей. На "внешнем виде" файла при просмотре браузером это никак не отражается, а его размер возрастает в четыре-пять раз. Соответственно растет и время загрузки.

В некоторых редакторах это свойство настраиваемое, в некоторых зашито жестко.

Редакторы HTML-текстов

Большинство таких редакторов вполне справляются с возложенными на них задачами. Мне лично очень понравились [HotDog](#) и [Ken Nesbitt Web Editor](#). Можете выгрузить 30-дневные тестовые версии и попробовать сами.

Конвертеры документов текстовых процессоров в HTML

Internet Assistant for Microsoft Word

Можно выгрузить бесплатно с WWW-сервера компании Microsoft (<http://www.microsoft.com/msword/internet/ia/>)

Превращает *Microsoft Word* версии 6.1 и выше в браузер и редактор HTML. Некорректно отображает таблицы и сохраняет русские буквы в виде &-последовательностей. От других людей мне доводилось слышать, что вторая проблема была решена в версии для *Word 7.0*. Так ли это на самом деле, сказать не берусь, поскольку все еще работаю с *Word 6.1*.

GT_HTML

Можно выгрузить бесплатно с http://www.gatech.edu/word_html/.

Понравилось, особенно после в целом негативного опыта работы с *Internet Assistant for Microsoft Word*. Требуется несколько большего объема рутинной работы, чем *Internet Assistant*, но и работает гораздо более корректно, включая воспроизведение таблиц. Существуют версии для *Microsoft Word 2.0*, *Microsoft Word 6.0* и *Microsoft Word for Macintosh*.

Другие

Краткую информацию на английском языке о некоторых других конвертерах, а также о том, где они лежат, можно найти на любом поисковом сервере. Ниже даны две ссылки на сервер Yahoo!:

- [Computers and Internet: Software: Internet: World Wide Web: HTML Converters](#)
- [Business and Economy: Companies: Computers: Software: Internet: Shareware: HTML Converters](#)

Графические редакторы

Скажу Вам сразу — мой опыт в этой области весьма ограничен. Компьютерная графика — это пограничная область между программированием и живописью (в тех случаях, конечно, когда она делается хорошо). В программировании я немного разбираюсь, а вот в живописи отнюдь не силен. Тем не менее, краткая информация о программном обеспечении может оказаться уместной.

Paint Shop Pro

Весьма симпатичный графический редактор. Поддерживает множество форматов графических файлов, в том числе и уже знакомые нам GIF и JPEG. Бесплатно распространяется в виде полностью функциональной версии для 30-дневных испытаний без механизма самоуничтожения. Существуют версии для *Windows 3.xx* и *Windows 95*. Можно выгрузить бесплатно с сервера компании-разработчика: <http://www.jasc.com/pspdl.html>.

GIF Construction Set

Программное обеспечение для создания подвижных изображений в формате GIF. Распространяется бесплатно. Существуют версии для *Windows 3.xx* и *Windows 95*. Можно выгрузить бесплатно с сервера компании-разработчика: <http://www.mindworkshop.com/alchemy/gifcon.html>.

Где можно бесплатно разместить свою страничку

Существует достаточно серверов, владельцы которых готовы позволить всем желающим размещать на них свои документы бесплатно. Для этого необходимо только знать английский язык в степени, достаточной для того, чтобы правильно заполнить регистрационную форму на сервере. Необходимо также уметь пользоваться программным обеспечением для загрузки файлов на удаленный сервер (CuteFTP, WS_FTP или аналогичным).

В этом смысле исключением выглядят серверы <http://www.halyava.ru/> и <http://www.i-connect.ru/>. Вся информация там на русском языке, так что нет смысла пересказывать. Посмотрите сами.

Информацию о некоторых других бесплатных серверах можно найти на поисковой системе [Yahoo](#). Информация там не всегда самая новая, так что настоятельно рекомендую посетить [страничку Вадима Хомахи](#), на которой такой информации много, свежей, проверенной и на русском языке.

Несколько слов в заключение

Надеюсь, что Вы смогли узнать что-то новое и полезное из этого руководства. Логичный вопрос теперь — куда двигаться дальше? Неплохая коллекция ссылок на русскую документацию по HTML и смежным вопросам есть на сервере ["Махаон"](#).

ДЕМОНСТРАЦИОННЫЕ МАТЕРИАЛЫ

(БЕЗ КОМЕНТАРИЕВ)

Обобщенный критерий оценки инвестиций в варианты
ДЕЯТЕЛЬНО-ТЕХНО-ПРИРОДНЫХ СИСТЕМ

$$\begin{aligned}
 & \sum_{m=1}^M \sum_{T=0}^A \left[K_D \left(K_{\text{вмтп}}^{\text{ИС}} + K_{\text{вмтп}}^{\text{Мо}} + K_{\text{вмтп}}^{\text{Ин}} + K_{\text{вмтп}}^{\text{Уп}} + K_{\text{вмтп}}^{\text{Ср1}} + K_{\text{вмтп}}^{\text{Че}} \right) \right] + \\
 & + \sum_{m=1}^M \sum_{T=0}^B \left[K_D \left(T_{\text{вмтп}}^{\text{ИС}} + T_{\text{вмтп}}^{\text{Мо}} + T_{\text{вмтп}}^{\text{Ин}} + T_{\text{вмтп}}^{\text{Уп}} + T_{\text{вмтп}}^{\text{Ср1}} + T_{\text{вмтп}}^{\text{Че}} \right) \right] + \\
 & + \sum_{j=1}^J \sum_{T=0}^A K_D \left(M_{\text{aj втп}} - \min_{\text{втп}} M_{\text{aj втп}} \right) \mathcal{E}_{\text{ор}}^j + \sum_{T=0}^A K_D \left(V_{\text{р втп}} - \min_{\text{втп}} V_{\text{р втп}} \right) \mathcal{E}_{\text{ор}}^{\text{вр}} + \\
 & + \sum_{T=0}^A \sum_{i=0}^I K_D \left(\frac{\max}{\text{втп}} \text{ПП}_{i \text{ втп}} - \text{ПП}_{i \text{ втп}} \right) \mathcal{Z}_i + \\
 & + \sum_{T=0}^A \sum_{k=1}^K K_D \left(O_{\text{вмтп}}^{\text{ТВ}} - \min_{\text{МТП}} O_{\text{вмтп}}^{\text{ТВ}} \right) \text{П}_k^{\text{ТВ}} \rightarrow \min.
 \end{aligned}$$

Здесь первые суммы учитывают капитальные вложения ($K_{\text{в}}$) в инженерную систему (ИС), модель (Мо), информацию (Ин), управление (Уп), частично в среду (Ср1) и человека (Че) – подготовка специалистов; вторые суммы учитывают текущие затраты, соответственно в ИС, Мо, Ин, Уп, Ср1 и Че – повышение квалификации. Индексы суммирования обозначают: m – номер мероприятия; v – номер варианта; l – последовательность осуществления мероприятий; $T_{\text{вм}}^n$ – время начала осуществления m -го мероприятия в v -м варианте для l -й последовательности; $\Delta T_{\text{вм}}^n$ – затраты времени на осуществление m -го мероприятия в v -м варианте для l -й последовательности ($a = T_{\text{вм}}^n$; $A = T_{\text{вм}}^n + \Delta T_{\text{вм}}^n$) – коэффициент дисконтирования затрат и результатов; $K_{\text{вмтп}}$ – капитальные вложения на осуществление m -го мероприятия в v -м варианте в t -м году для l -й последовательности; $T_{\text{вмтп}}$ – текущие затраты после осуществления m -го мероприятия в v -м варианте в t -м году для l -й последовательности; T – расчетный срок, данный сроку создания «комплекса» плюс два срока окупаемости; $\text{ПП}_{i \text{ втп}}$ – объем потребительского продукта i -го вида ($i = 1, \dots, I$) в v -м варианте в t -м году для l -й последовательности; \mathcal{Z}_i – замыкающие затраты на продукцию i -го вида; $M_{\text{aj втп}}$ – объем отвлекаемых материальных ресурсов j -го вида ($j = 1, \dots, J$) в v -м варианте t -м году для l -й последовательности; $\mathcal{E}_{\text{ор}}^j$ – экономическая оценка ресурса j -го вида; $V_{\text{р втп}}$ – время, используемое в v -м варианте в t -м году для l -й последовательности; $\mathcal{E}_{\text{ор}}^{\text{вр}}$ – экономическая оценка времени; $O_{\text{вмтп}}^{\text{ТВ}}$ – объем технологического выброса k -го вида ($k = 1, \dots, K$) в v -м варианте в t -м году для l -й последовательности; $\text{П}_k^{\text{ТВ}}$ – плата за единицу объема технологического выброса k -го вида

$$B = T_{\text{в}}^n + \Delta T_{\text{в}}^n; \quad B = T_{\text{в}}^n (m+1) + \Delta T_{\text{в}}^n (m+1).$$

Если капиталовложения и текущие затраты в компоненты «комплекса» осуществляются одновременно, то суммирование каждого слагаемого в квадратных скобках производится отдельно в соответствующих временных интервалах a , A и b , B . Может также оказаться, что отрасли, входящие в АПК в качестве подсистем, будут иметь отличные числовые значения нормативного коэффициента E .

СОДЕРЖАНИЕ	
1. ОРГАНИЗАЦИЯ СИСТЕМНО - СТРУКТУРНЫХ ИССЛЕДОВАНИЙ	5
2. В основе концепции лежат следующие системные принципы.	17
3. Методологические основы комплексной мелиорации агроландшафта с позиций деятельно – техно - природных систем, организующих сельскохозяйственную и мелиоративную деятельность.	18
3.1. Обобщенный взгляд на деятельно – техно – природные системы.	24
3.2. В основе концепции лежат следующие системно - структурные взгляды системного подхода	
3.3. В основе концепции лежат следующие содержательные положения.	
4. Разработки и внедрения экспертных систем в области мелиораций.	
4.1. <i>Экспертные системы как эффективное средство решения не формализуемых и трудно формализуемых задач.</i>	
4.1.1. <i>Традиционные средства программирования и не формализуемые и трудно формализуемые задачи.</i>	
4.1.2. Структура и состав экспертной системы.	
4.1.3. Разработка и внедрение экспертных систем.	
4.2. Принципы построения базы знаний в системе производства продукции на мелиорируемых землях.	
4.3. Построение категориально-понятийного пространства.	
4.3.1. Структура элемента базы знаний.	
4.4. Структура базы знаний.	
4.5. Система управления базой знаний «КАРТА ЗНАНИЙ».	41
4.5.1. Общие сведения о программном комплексе.	
4.5.2. Начало работы и парольная система.	
4.5.3. Создание логической структуры БЗ. Работа с деревьями КПП.	
4.5.4. Матричное представление КПП.	
4.5.5. Ввод и корректировка элементов знаний.	
4.5.6. Ввод и редактирование текста.	
4.5.7. Определение имени файла.	
4.6. Структура программного обеспечения системы.	
4.7. Структура базы знаний.	
5. Автоматизированная система "Карта знаний науки"	
5.1. Построение категориально-понятийного пространства.	48
5.2. Структура элемента базы знаний.	
5.3. Описание программы.	
5.4. Основные функции программы.	
5.4.1. Работа с деревьями.	60
5.4.2. Работа со списком рефератов.	61
5.4.3. Ввод нового реферата.	
6. РАЗРАБОТКА БАЗЫ ЗНАНИЙ С ИСПОЛЬЗОВАНИЕМ АППАРАТА КАТЕГОРИАЛЬНО-ПОНЯТИЙНОГО ПРОСТРАНСТВА.	
6.1. Построение категориально-понятийного пространства.	63
6.2. Структура элемента базы знаний.	
7. ПОЛЬЗОВАТЕЛЮ АВТОМАТИЗИРОВАННОЙ СИСТЕМОЙ "КАРТА ЗНАНИЙ"	
7. 1. Установка параметров БЗ.	
7. 2. Реорганизация базы знаний	
7. 3. Слияние баз знаний.	66
7. 4. Поиск по ключевым словам.	
8. HTML	
8.1. Вводные замечания	
• Для чего нужно это руководство?	
• Что такое HTML?	
• Что Вам понадобится для освоения HTML?	67
• Необходимое отступление о редакторах HTML	

<u>8.2. Наш первый HTML-документ</u>	
<ul style="list-style-type: none"> • Как устроен HTML-документ • Обязательные метки <ul style="list-style-type: none"> ○ <html> и </html> ○ <head> и </head> ○ <title> и </title> ○ <body> и </body> ○ <H1> и </H1> ‡ <H6> и </H6> ○ <P> и </P> 	71
<u>8.3. Внутри абзаца</u>	
<ul style="list-style-type: none"> • Непарные метки <ul style="list-style-type: none"> ○ Конец строки:
 ○ Горизонтальная линия: <HR> ○ &-последовательности ○ Комментарии • Форматирование шрифта <ul style="list-style-type: none"> ○ Физические стили ○ Логические стили 	78 80
<u>8.4. Организация текста внутри документа</u>	
<ul style="list-style-type: none"> • Ненумерованные списки: и • Нумерованные списки: и • Списки определений: <DL> и </DL> • Вложенные списки • Форматированный текст: <PRE> и </PRE> • Текст с отступом: <BLOCKQUOTE> и </BLOCKQUOTE> 	86 91
<u>8.5. За пределами HTML-файла</u>	
<ul style="list-style-type: none"> • Связывание • Изображения в HTML-документе 	
<u>8.6. Тайное становится явным, или Видимое действие невидимых меток</u>	
<ul style="list-style-type: none"> • Цветовая гамма HTML-документа • Заголовок HTML-документа: что в нем может быть интересного? 	
<u>8.7. Таблицы</u>	
<ul style="list-style-type: none"> • Для чего нужны таблицы? • Как устроена таблица 	
<u>8.8. Формы</u>	
<ul style="list-style-type: none"> • Для чего нужны формы? • Как устроена форма <ul style="list-style-type: none"> ○ Простейшая форма ○ Как форма собирает данные • Маленькая хитрость, или как отправить форму почтой 	
<u>8.9. Некоторые (предположительно полезные) советы по разработке и размещению HTML-страниц</u>	
<ul style="list-style-type: none"> • Правила хорошего тона при разработке HTML-документов • Инструменты разработчика: пристрастные советы. 	
9. ДЕМОНСТРАЦИОННЫЕ МАТЕРИАЛЫ (БЕЗ КОМЕНТАРИЕВ)	
Разработаны образцы меню, программ и таблиц в HTML-формате и др. для представления компонентов «ДЕЯТЕЛЬНО – ТЕХНО – ПРИРОДНЫХ СИСТЕМ»	

Учебное пособие

Леонид Мечеславович РЕКС
АКАДЕМИК Российской Академии Естественных Наук,
ПРОФЕССОР, ЛАУРЕАТ ПРЕМИИ СОВ. МИН СССР, ДОКТОР ТЕХНИЧЕСКИХ НАУК

СИСТЕМНЫЕ ИССЛЕДОВАНИЯ И ИНФОРМАТИКА
ДЕЯТЕЛЬНО – ТЕХНО – ПРИРОДНЫХ СИСТЕМ
УЧЕБНОЕ ПОСОБИЕ
ЧАСТЬ 1.

Печатается в авторской редакции
Компьютерная верстка Л. М. Рекс

Подписано в печать 15.03.2004 г. Тираж- 300 экз. Формат бумаги 60x84/16 Заказ № 101

Отпечатано в лаборатории множительной техники Московского государственного
университета природообустройства (МГУП).
127550, Москва, ул. Прянишникова 19.

