**Содержание**

1. Принципы построения управленческих информационных систем.

2. Инструментальные средства для работы пользователя. Работа с окнами и формулами

Список литературы

1. **Принципы построения управленческих информационных систем**

Эволюция информационных систем прошла путь длиной в 35 лет. С развитием компьютерной техники, программных средств, методов управления информацией менялся и смысл, вкладываемый в это понятие – теперь уже никто не назовет электронную таблицу с калькулятором таким громким именем. Современные информационные системы являются сложными интегрированными комплексами, которые включают в себя модули, отвечающие практически за все механизмы работы современного предприятия. Информационная система – это набор механизмов, методов и алгоритмов, направленных на поддержку жизненного цикла информации и включающих три основных процесса: обработку данных, управление информацией и управление знаниями. С точки зрения программных технологий информационная система – это не один, и даже не несколько программных комплексов. Можно построить структурную модель информационной системы (см. рис.11), выделив ее основные компоненты, которые содержат программные модули определенного класса.

Автоматизированные системы управления технологическими процессами

Системы планирования ресурсов предприятия (ERP)

Системы анализа и принятия решений (DSS)

Система управления знаниями

Хранилище данных,

документов, приказов, справочников

Шлюз для обмена с внешними информационными системами

Внешняя информационная система

Рис. 11. Структурная схема современной информационной системы

Самым нижним уровнем информационной системы является хранилище, в котором содержится вся интеллектуальная собственность предприятия. Это могут быть документы, справочники, структурные таблицы, деловые правила, описание процессов. Прямого доступа к хранилищу быть не должно, как для пользователей, так и для различных систем предприятия. Прямой доступ имеет лишь система управления знаниями, которая служит своего рода шлюзом для остальных систем и формирует информационное окружение предприятия. Система управления знаниями объединяет идеи, знания, содержание документов и деловые правила, автоматизируя процессы, базирующиеся на знаниях, как внутри предприятия, так и между разными организациями. Для этого нужен шлюз, позволяющий производить обмен данными с внешними системами. Это необходимое условие, так как современные процессы направлены на объединение предприятий в крупные концерны и очевидно, что передача знаний очень важна. Например, системы планирования ресурсов предприятия (ERP – enterprise resource planning) не могут работать независимо – процессы, связанные с управлением финансами, складами, человеческими ресурсами, используют уже накопленные знания и приносят новые.

Также важно выделить класс систем анализа и принятия решений (DSS–decision support system), без которого жизненный цикл информации не будет завершен. В современных организациях интеллектуальный анализ данных становится все более важной задачей. Связано это с необходимостью аналитической обработки больших объемов информации, накопившейся в хранилищах. Такие системы помогают найти новые знания, выявить недостатки и слабые места информационной системы, оценить эффективность тех или иных процессов, установить новые информационные взаимосвязи.

Очень часто говорят, что такой класс систем должен работать непосредственно с хранилищем, поскольку обработке подлежат содержащиеся в нем данные. Теоретически это верно, но на практике такое невозможно – любые изменения в содержимом хранилища, процессах, правилах и взаимосвязях могут и должны производиться системой управления знаниями. Тогда DSS – системам не придется задумываться над тем, в каком формате хранятся данные, и главное, что любое изменение информации будет немедленно влиять на взаимосвязи и процессы, в которых она принимает участие.

## Понятие – База данных

В информационной системе с использованием технологии баз данных решается задача информационного моделирования какой-либо предметной области (ПО) или её фрагмента. Основа УИС, объект ее обработки – база данных.

Что такое база данных (БД)? В широком смысле слова можно сказать, что БД – это совокупность сведений о конкретных объектах реального мира в какой-либо предметной области.

Основные черты концепции БД:

данные отделяются от программ, появляется специальная программная надстройка для управления данными, называемая системой управления базами данных (СУБД); СУБД управляет данными и служит посредником между ними и программами, они упрощаются, освобождаются от функций структуризации, хранения и поиска данных;

появляются стандартизированные данные о фактографических данных – метаданные, управляемые СУБД; метаданные описывают информационные параметры и взаимосвязи фактографических данных о ПО;

СУБД совместно с метаданными представляет собой стандартизированное инструментальное средство для моделирования ПО различной природы;

происходит централизация (интеграция) данных, их многоаспектное использование для различных приложений, что сокращает избыточность данных, позволяет обеспечить более высокий уровень достоверности данных и оптимизировать различные процедуры ведения и использования БД.

Основными функциями СУБД являются:

управление данными во внешней памяти;

управление буферами оперативной памяти;

управление транзакциями;

журнализация;

поддержка языков БД.

## Общие сведения о проектировании БД

Процесс проектирования базы данных выполняется поэтапно, а этапы в основном соответствуют разновидностям моделей программного обеспечения при движении от более абстрактных к более конкретным с датологической точки зрения: концептуальной инфологической модели и двух датологических, логического уровня и внутреннего уровня. Построению этих моделей предшествует изучение предметной области.

Таким образом, выделяются следующие четыре этапа проектирования:

* обследование ПО, формирование и анализ требований;
* инфологическое проектирование;
* логическое проектирование;
* внутреннее (физическое) проектирование.

Каждому из этапов соответствуют свои принципы, методы, приемы.

Основное содержание первого этапа: сбор сведений о сущностях, их свойствах и взаимоотношениях в ПО; о процедурах, связанных с объектами ПО; о требованиях по объемам информации в БД, быстродействию, пользователях и т.п.

Для специальных ПО приходится общаться со специалистами и экспертами, может использоваться методология проведения экспертных оценок и обработки их результатов. Для обследования и описания ПО существует целый ряд подробно разработанных методик, которые предлагают виды проработанных таблиц для заполнения, вопросники и т.п. вспомогательные средства. Это облегчает и стандартизирует работу по обследованию ПО, позволяет сократить время изучения.

На этапе концептуального, инфологического проектирования разрабатывается концептуальная схема БД. Главные проблемы заключаются в структуризации информационной анархии, полученной в результате сбора информации о ПО, в решении вопросов:

объединения информации из различных фрагментов ПО;

выделения объектов группировкой атрибутов (при этом семантические связи разделяются на внутренние, между атрибутами в составе объектов, и внешние – между сущностями);

выбора ключей;

учета и отображения в составе связей структурных и запросных связей.

Все это решается неоднозначно, но от рационального решения этих вопросов сильно зависит качество БД. Чаще всего при решении указанных вопросов используется терминология и приемы, разработанные в рамках реляционной модели данных (терминология отношений, методы нормализации отношений). Делаются попытки создать в этой сфере автоматизированные системы, подобие САПР.

Существуют два подхода к ПО:

исторически первый (как более простой и быстрый) основан на интегрировании представлений о ПО пользователей информации;

второй базируется на представлениях об объективно (независимо от пользователей) существующей ПО, с присущей ей семантикой.

Современная точка зрения требует сочетания обоих представлений. Без учета второго подхода не будет достаточной гибкости и способности к адаптации при корректировке пользовательских потребностей.

На первом и втором этапах используются такие общеметодологические принципы, как приемы классификационного анализа, принципы системного анализа, принципы анализа и синтеза.

При объединении локальных представлений о фрагментах ПО в единое концептуальное представление используются три основных принципа: идентичности, агрегации, обобщения.

На третьем этапе, этапе логического проектирования, выбирается логический тип модели данных (например из классических: сетевой, иерархический, реляционный) и конкретная СУБД этого типа. Производится отображение концептуальной схемы на выбранную модель с учетом ограничений конкретной СУБД.

На четвертом этапе, при физическом проектировании, решаются вопросы конкретного использования выбранной СУБД для наиболее эффективного выполнения запросов. Здесь выбирается способ организации файлов, методы доступа, способы организации и размеры буферов и блоков, способы индексирования и прочее. Обычно СУБД решает эти вопросы автоматически, по умолчанию, но эти решения могут быть изменены с помощью настроек и специальных процедур.

1. **Инструментальные средства для работы пользователя. Работа с окнами и формулами**

Одним из наиболее распространненных инструметов работы пользователя являются электронные таблицы. В силу того, что программный продукт Excel фирмы Microsoft обладает очень широким набором функциональных возможностей, а также тесно интегрирован со всей линейкой Microsoft Office он нашел очень широкое применение на рабочих местах управленцев всех уровней.

MS Excel за свою мощь и универсальность получил название табличного процессора. И в этом состоит основная идея программы - Excel хранит информацию, организуя ее с помощью таблиц. Однако хранение информации - это еще половина дела. Второе и главное преимущество Excel - это многочисленные возможности для анализа этой информации: проведения всевозможных вычислений, построения наглядных зависимостей. Любому, кто сталкивался с необходимостью анализировать какую либо информацию известно, что даже если нет необходимости в выполнении вычислений, иногда имеет смысл располагать информацию в табличной форме, чтобы подчеркнув тем самым некую систему, облегчить ее субъективное восприятие. Поэтому и сама программа Excel ориентированна на самый широкий круг пользователей и применима при решении широчайшего спектра задач.

## Работа с окнами

### Закрепление заголовков

Чтобы заголовки таблицы всегда оставались на экране. Меню "Окно" - пункт "Закрепить области". Можно выделить строку или столбец для определения их как заголовка. Можно сделать и то и другое. Для этого надо выбрать одну ячейку. Курсор остается ниже и правее закрепленной области. Последовательно закрепить как строку так и столбец. Затем, закрепить и то и другое одновременно. Снять закрепление областей.

### Операции с окнами

Разделение - Меню "Окно" - команда "Разделить". "Разделяет" окно на четыре независимых области, размеры каждой из которых могут быть изменены за счет соседней. Таким образом одновременно на экране могут изображаться 4 отдаленные области одного листа.

Новое окно - Меню "Окно" - команда "Новое". На экране появляется второе (и более) окно, что позволит одновременно просматривать на экране два и более листов одной книги.

Расположить окна - Меню "Окно" - команда "Расположить…". Позволяет расположить на экране открытые книги (окна) в соответствии с вашим желанием.

### Найти и заменить

 Дает возможность найти на листе различные сочетания текстовых и числовых значений. Меню "Правка" - "Найти". Диалог "Найти". Строка "Что". Список "Просматривать" ("По строкам", "По столбцам"). Список "Область поиска" (Формулы, значения, примечания). Флажки "Учитывать регистр" и "Ячейка целиком".

Задание. Ввести в строке "Что" значение одной из ячеек на листе. "Найти" ее. Ввести в "Что" знак "\*". Программа будет последовательно проходить по всем ячейкам листа, содержащим какие-либо данные.

Заменить. Дает возможность заменять найденные на листе сочетания текстовых и числовых значений. Меню "Правка" - "Заменить". Кроме уже описанных в "Найти..." возможностей добавляется строка "Заменить на...".

### Прогрессия

Несколько последовательно расположенных ячеек в столбце или строке могут быть автоматически заполнены неким числовым рядом. Этот ряд может быть назван прогрессией. Перед выполнением операции необходимо выделить диапазон в строке или столбце предназначенный для заполнения. В начальной ячейке этого диапазона должно помещаться число, которое явится первым значением числового ряда. Меню "Правка", "Заполнить" - "Прогрессия". Группа "Расположение" - "По строкам", "По столбцам". Тип прогрессии - "Арифметическая", "Геометрическая", "Автозаполнение". "Шаг" и "Предельное значение". Флажок "Автоматическое определение шага" работает тогда, когда в первых ячейках зоны заполнения присутствует как минимум два значения числового ряда.

Задание. Выполнить арифметическую прогрессию. Затем Геометрическую прогрессию с шагом "2".

### Сортировка

Сортировка используется для упорядочивания строк в списке в соответствии с содержимым отдельного столбца. Сортировке подлежат две категории данных - текстовые и числовые. После сортировки ячеек, содержащих формулы, эти формулы могут возвращать неверные результаты. Если сортировке подлежат данные в столбце, который связан с другими столбцами по строкам, то выделять необходимо все эти столбцы. Выделение столбцов начинается либо слева направо, либо справа налево - в зависимости от того, по какому из столбцов (правому или левому) будет производится сортировка. Если нет необходимости в задании особых условий сортировки, то можно пользоваться кнопками "Сортировка по возрастанию" и "... убыванию", в панели инструментов "Стандартная". Расширенные возможности для сортировки предоставляются в меню "данные" - командой "Сортировка". Диалог сортировка. Переключатели "Идентифицировать столбцы по…"- принимать или нет верхнюю строку сортируемой области за заголовок и сортировать ли ее. Идея диалога "Сортировка" - можно осуществить последовательную сортировку по нескольким независимо определяемым критериям.

Задание. Ввести две колонки данных: буквенные и числовые. Сортировать первый в порядке возрастания, второй - в порядке убывания относительно первого.

### Автофильтр

Идея принципа фильтрации состоит в том, что в какой-то момент времени в таблице визуально отображаются лишь те данные, которые удовлетворяют каким либо заданным критериям. Эти критерии могут содержатся в ячейках самой таблицы, либо являться условием, указываемым пользователем. После создания таблицы горизонтальным выделением следует определить ее заголовок. Меню "Данные" - подменю "Фильтр" - команда "Автофильтр". После ее выбора в углах ячеек заголовка появятся треугольнички раскрывающихся списков; из данных списков производится выбор значения. При необходимости задать какое либо условие отбора, для отображения диапазона значений столбца, оно может быть сформировано с помощи команды "условие". В диалоге "Пользовательский автофильтр" пользователю предоставляется возможность ввести границы диапазона выводимых значений с помощью знаков < , > , = , и их комбинаций.

Примечание: если автофильтр включен, то в таблице могут отображаться не все строки таблицы. Тогда номера строк окрашиваются синим цветом, одновременно с этим синим окрашивается и стрелочка списка в столбце, где применена автофильтрация. Для выхода из режима автофильтрации следует либо снять все условия, либо отменить автофильтр, сняв "птичку" против команды "Автофильтр" в меню "данные".

Задание. Создать таблицу из двух столбцов и нескольких строк. Применить автофильтр. Выбрать из списка "условие". Ввести условие. Отменить автофильтр.

## Ввод ссылок и формул

### Ввод формул

В одной ячейке может помещаться не более одной формулы. Формула может вводиться как непосредственно в ячейку так и через строку формул. Любая формула начинается со знака "=", содержит ссылки на имена ячеек, содержащих исходные данные для вычислений и арифметические операторы. Вместо ссылок возможен ввод числовых констант непосредственно в формулу, однако делать это следует лишь тогда, когда эти числа не предполагается изменять в последующем. Условно формулы можно подразделить на несколько типов.

Простые формулы. Они содержат ссылки на ячейки и арифметические операторы. Как известно, любая, как простая, так и сложная формула, в конечном счете, состоит из четырех арифметических действий ( +; -; \*; /.) поэтому можно сказать, что простые формулы служат базой для написания сложных.

Усложненные формулы (функции). Они состоят из нескольких (многих) арифметических действий и обрабатывают несколько переменных. С точки зрения технологии ввода формул любая функция может быть написана вручную. Однако для упрощения этого процесса существует мастер функций. Он позволяет вводить ряд широко используемых функций, записывая в ячейку лишь их условное название и ссылки на влияющие ячейки.

Частным случаем формулы является также ссылка. Ссылка это средство копирования содержимого одной ячейки в другую с сохранением обновляемой связи с источником данных. Как и формула ссылка начинается со знака "=", однако не содержит арифметических операторов. Ссылки могут быть внешними, т.е. указывать на ячейки на других листах или в других книгах.

Мастер функций: Меню "Вставка" - команда "Функция". Также мастер функций вызывается кнопкой в строке формул или в панели инструментов "Стандартная". По своему назначению и принадлежности функции тематически объединяются в категории. Переключаясь по категориям пользователь открывает список доступных функций. По каждой из них он может получить подробную справку, нажав кнопку "Справка". Для создания расчетной формулы в выделенной ячейке пользователь должен "ответить" на вопросы мастера о том, в каких ячейках листа помещаются исходные данные для вычислений.

### Абсолютные и относительные ссылки

При написании формул по умолчанию используются относительные ссылки. Разница между ними проявляется при последующем копировании формул из одной ячейки в другую. При перемещении или копировании абсолютные ссылки в формулах не изменяются, а относительные ссылки автоматически обновляются в зависимости от нового положения, таким образом, что они начинают указывать на разные ячейки. Взаимосвязь между ячейками новых формул и новыми ссылками подобна взаимосвязи ячейки исходной формулы и ссылок, которые в исходной формуле используются. При копировании формул с относительными ссылками, они изменяются таким образом, чтобы сохранить "геометрическую" привязку формулы к ячейкам с исходными данными. Примеры : =$A$1\*$B$1- абсолютная; =A1\*B1 - относительная; =$A1\*$B1 - смешанная. Нажимать F4 для циклического изменения типа ссылки (относительная, абсолютная, смешанная) до тех пор, пока не выберите необходимый тип.

Пример написания формулы на базе функции "Если". Постановка задачи: Написать формулу для расчета премиальных выплат за перевыполнение плана. Ход решения: Ячейка А15 содержит число, большее чем 0, которое говорит о том, насколько процентов был выполнен план. Если план был выполнен менее, чем на 100 процентов, то премии не полагается - формула должна возвращать ноль. Если план выполнен на 100 и более процентов, формула должна возвращать значение его действительного выполнения (копировать ячейку А15). При написании формулы возможно применение мастера функций.

### Циклическая ссылка

Нередко пользователи получают сообщение об ошибке, где говорится о какой-то циклической ссылке. В действительности, циклические ссылки не являются ошибочными, однако по умолчанию запрещены. Циклическая ссылка - это замкнутая последовательность ячеек, последовательно ссылающихся друг на друга. Для разрешения применения циклических ссылок следует выбрать пункт "Итерации" на вкладке "Вычисления" в меню "Сервис" команда "Параметры".

### Зависимости (Трассировка)

Панель инструментов "Зависимости". Меню "Сервис" - пункт "Зависимости". Зависимые и зависящие ячейки. Трассировка. Ячейки содержащие исходные данные для вычислений являются влияющими (зависящими). Ячейки, содержащие результаты вычислений - зависимыми. Ячейки, содержащие промежуточные результаты вычислений будут как зависимыми так и влияющими. Последовательность зависимостей можно проследить, установив стрелки. Неубранные стрелки могут быть выведены на печать. Если в вычислениях есть ошибка, она может быть найдена. Стрелки, идущие от ячейки с неверной формулой окажутся красного цвета. Такое может возникать, например, при делении на ноль. Ячейка возвращает сообщение типа "Dev/0" и генерирует красные трассы. Деление на ноль запрещено, однако сам ноль не является ошибочным значением. Ошибки следует искать либо в формуле, либо в ячейках, на которые она ссылается.

### Пересчет формул

Пересчет формул может быть организован автоматический или ручной.. При ручном пересчете после ввода новых исходных значений следует каждый раз нажимать клавишу F9. Если это не удобно выполнить следующие действия: В меню "Сервис" выберете вкладку "Параметры". На вкладке "Вычисления" выберете кнопку "Производить пересчет автоматически".

**Список литературы**

1.Конспект лекций по дисциплине «Управлінські інформаційні системи в аналізі і аудиті», СидоренкоИ.И.