Министерство образования Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б.С. Галущака»

**Реферат**

Объектная модель работы с данными

ОП.09 Стандартизация, сертификация и техническое документирование

НАТКиГ.421221.010.000ПЗ

Разработала:

студентка группы ПР-21.101,

Коршунова А. А.

Проверила:

Терехова А. А.

2023

**Содержание**

[Введение 3](#_Toc156294349)

[1 Понятие объектной модели 4](#_Toc156294350)

[2 Преимущества объектной модели 6](#_Toc156294351)

[3 Ограничения объектной модели 7](#_Toc156294352)

[4 Объектная модель данных в программе "1С: Предприятие" 8](#_Toc156294353)

[Заключение 14](#_Toc156294354)

[Список источников 15](#_Toc156294355)

# Введение

В современном мире объемы данных постоянно растут, и эффективная работа с ними становится все более важной задачей. Объектная модель работы с данными представляет собой один из основных подходов к организации и управлению информацией в компьютерных системах. Этот подход основан на использовании объектно-ориентированного программирования для представления данных в виде объектов с определенными свойствами и методами. В данном реферате будут приведены преимущества и недостатки объектной модели, а также рассмотрены объекты и их структура.

# 1 Понятие объектной модели

Объектная модель – это способ представления данных и их связей в базе данных с использованием объектно-ориентированного подхода. В объектной модели данные представлены в виде объектов, которые имеют свои свойства (атрибуты) и методы (операции).

В объектной модели данные представлены в виде объектов, которые имеют свои свойства и методы. Свойства объекта представляют его состояние, а методы – его поведение. Объекты могут взаимодействовать друг с другом, обмениваясь сообщениями и вызывая методы других объектов.

Основные понятия объектной модели:

**—** класс: это шаблон или описание, по которому создаются объекты. Класс определяет набор свойств и методов, которые будут у объектов этого класса;

— объект**:** это экземпляр класса. Он создается на основе определенного класса и имеет свои уникальные значения свойств;

— свойство: это характеристика объекта, которая описывает его состояние. Свойства могут быть различных типов данных, таких как числа, строки, булевы значения и т.д;

**—** метод: это функция, которая определяет поведение объекта. Методы могут изменять состояние объекта, взаимодействовать с другими объектами и выполнять различные операции;

**—** наследование: это механизм, который позволяет создавать новые классы на основе уже существующих. Наследование позволяет наследовать свойства и методы родительского класса и добавлять к ним новые или изменять существующие;

— полиморфизм: это возможность объектов разных классов иметь одинаковые методы с одинаковыми именами, но с различной реализацией. Полиморфизм позволяет использовать объекты разных классов в общих операциях;

— инкапсуляция: это принцип, который позволяет скрыть внутреннюю реализацию объекта и предоставить только необходимый интерфейс для взаимодействия с ним. Инкапсуляция обеспечивает защиту данных и облегчает изменение внутренней реализации без влияния на внешний код.

# 2 Преимущества объектной модели

Объектная модель имеет ряд преимуществ, которые делают ее полезной и эффективной для разработки программного обеспечения. Вот некоторые из них:

— Модульность и повторное использование кода: объектная модель позволяет разбить программу на отдельные модули или классы, каждый из которых выполняет определенную функцию. Это делает код более организованным, понятным и легко поддерживаемым. Кроме того, благодаря наследованию и полиморфизму, можно повторно использовать уже написанный код, что экономит время и усилия разработчика;

— упрощение сложных систем: объектная модель позволяет абстрагироваться от сложности системы и представлять ее в виде набора объектов, которые взаимодействуют друг с другом. Это упрощает понимание и анализ системы, а также упрощает ее разработку и поддержку.

— гибкость и расширяемость: объектная модель обеспечивает гибкость и расширяемость программного обеспечения. Благодаря наследованию и полиморфизму, можно легко добавлять новые функции и изменять поведение существующих объектов без необходимости изменения всей системы. Это позволяет адаптировать программное обеспечение под новые требования и сценарии использования.

— улучшение тестирования: объектная модель делает код более понятным и структурированным, что упрощает его сопровождение и тестирование. Каждый объект выполняет определенную функцию, и его поведение можно легко проверить и протестировать независимо от других объектов. Это упрощает обнаружение и исправление ошибок, а также добавление новых функций без нарушения работоспособности системы.

# 3 Ограничения объектной модели

— Необходимость явного определения структуры данных: в объектной модели необходимо явно определить структуру данных, то есть классы и их атрибуты. Это требует дополнительной работы по проектированию и определению классов, что может быть сложным и затратным процессом;

— ограничения наследования: в объектной модели наследование может быть ограничено определенными правилами и ограничениями. Например, некоторые языки программирования могут запрещать множественное наследование или наследование от конкретных классов. Это может ограничить возможности моделирования и создания иерархий классов;

— проблемы согласованности данных: объектной модели может возникнуть проблема согласованности данных, когда данные в разных объектах могут быть несогласованными или несогласованными с базой данных. Например, если объекты имеют копии данных, то изменение одного объекта может не отразиться на других объектах или в базе данных;

— проблемы производительности: использование объектной модели может привести к проблемам производительности из-за дополнительных накладных расходов на создание и управление объектами. Кроме того, объектная модель может быть менее эффективной при работе с большими объемами данных или при выполнении сложных операций;

— сложность взаимодействия с другими системами: взаимодействие объектной модели с другими системами или базами данных может быть сложным из-за различий в структуре данных и языках программирования. Необходимо разработать специальные механизмы и протоколы для обмена данными между объектной моделью и другими системами.

# 4 Объектная модель данных в программе "1С: Предприятие"

**4.1 Объекты**

Объектная модель представляет структуру и состав предметной области. Она состоит из основных объектов, предназначенных для моделирования документооборота и учета в целом. Использует важнейшие основные объекты: документ, справочник, регистр, отчет.

*Документ* – объект, предназначенный для представления первичной информации о совершенных хозяйственных операциях или о событиях, произошедших в жизни организации вообще. Документ обладает уникальной способностью – проведением. Это означает, что событие, которое документ отражает, повлияло на состояние учета. Факт проведения означает, что документ обработан полностью. Еще одна важная особенность документа – его привязка к шкале времени. Это позволяет отслеживать последовательность событий.

*Отчет* – объект, предназначенный для представления алгоритмов и инструментов получения выходных данных. Отчеты обобщают, формируют и выводят в заранее заданной форме учетную информацию.

Из перечисленных основных объектов уже можно построить завершенную информационную систему (рис.3.1).



Рис.1 Простейшая информационная система на основе документов и отчетов

Однако, для целей практической разработки этих объектов недостаточно. Необходимо ввести еще два объекта, один из которых отвечает за организацию ввода данных, а второй за организацию хранения данных.

*Справочник* – объект, предназначенный для описания однородных данных, представленных в виде списков. Данные, хранимые в справочниках, можно считать условно-постоянными. Хотя пользователь имеет возможность редактировать содержимое справочников, происходит это сравнительно нечасто.

*Регистр* – объект, предназначенный для описания структур накопления данных. Регистр содержит измерения – свойства, обеспечивающие анализ накапливаемых данных в требуемых разрезах. Регистр содержит ресурсы – виды накапливаемых данных. Регистр, как правило, связан с документом и обеспечивает сохранение данных, изменяемых или создаваемых при проведении документа. С практической точки зрения регистры обеспечивают повышение производительности конфигурации, так как они реализуют механизм распределения во времени вычислительной нагрузки. Действительно, если в регистры помещается итоговая, суммирующая информация, к которой впоследствии происходит обращение, значит часть вычислительных затрат реализована заранее – при помещении данных в регистр. Дополнительное увеличение производительности системы обеспечивается за счет применения еще одного специализированного механизма регистров – виртуальных таблиц.



Рис.2 Схема взаимодействия основных объектов

Все основные объекты могут иметь набор дополнительных свойств, которые называются реквизитами. Набор реквизитов для каждого объекта отражает его специфические свойства и вообще говоря, может быть уникальным.

**4.2 Справочники**

Назначение основного объекта Справочник. Справочники предназначены для хранения условно-постоянных данных, т.е. данных, которые изменяются сравнительно редко, либо не изменяются вообще. Каждый справочник представляет собой список, который может быть структурирован линейно или иерархически. В зависимости от структуры, справочник может содержать или одни элементы или группы и элементы. У каждого элемента есть стандартные реквизиты: код (число или строка), наименование, родитель(имя группы или элемента верхнего уровня в иерархических справочниках), владелец (имя элемента владельца в подчиненном справочнике), ЭтоГруппа (булевского типа), ПометкаУдаления (булевского типа), Предопределенный (булевского типа), Ссылка (уникальное «внутреннее» поле для обращения к конкретному элементу).

Справочники могут иметь предопределенные элементы – создаваемые на этапе разработки и запрещенные для редактирования пользователями. Предопределенные элементы используется при необходимости обратиться к каким-либо элементам из программного кода.

Программные объекты справочников. С помощью встроенного языка можно выполнять различные операции над справочниками: создавать, изменять и удалять элементы, искать нужный элемент, перебирать элементы в цикле и т.д.

Работа со справочниками из программы производится с помощь нескольких объектов. Каждый объект имеет свое назначение, собственные свойства и методы. Ниже описываются эти объекты и взаимосвязи между ними:

СправочникиМенеджеробеспечивает доступ ко всем справочникам конфигурации. Свойства этого объекта совпадают с именами справочников и содержат объекты типа СправочникМенеджер.

СправочникМенеджеробеспечивает доступ к операциям над справочником как множеством элементов. Через методы этого объекта можно осуществлять поиск, получать выборку, создавать новые элементы или группы, обращаться к формам и макетам справочника. Например: Справочники.ФизическиеЛица.

СправочникОбъект.Предоставляет доступ к элементу с возможностью записи. Данный объект содержит методы, влияющие на элемент в базе данных, например, методы Записать и Удалить.

СправочникСсылка.Данный объект однозначно идентифицирует элемент (группу) справочника и позволяет обращаться к нему в режиме «только чтение». Через свойства и методы этого объекта можно прочитать реквизиты элемента (группы), обратиться к его табличным частям. Значение этого типа хранится в реквизитах, ссылающихся на элементы данного справочника, например, в реквизите Сотрудник документа ПриемНаРаботу хранится ссылка на конкретный элемент справочника Сотрудники.

**4.3 Документы**

Документы предназначены для хранения первичных данных о хозяйственных операциях. Последовательность документов, привязанных к шкале времени, позволяет восстановить последовательность событий. Поэтому каждый документ обладает двумя стандартными реквизитами – *датой* и *номером* документа. Кроме того, стандартными реквизитами документа являются *пометка удаления, ссылка, проведен.*

Стандартный реквизит Номер имеет уникальное значение в пределах заранее заданного для документа периода, например месяц, квартал, год. В начале следующего периода нумерация возобновляется.

Также возможны ситуации, когда единая нумерация применяется для разнотипных документов. В этом случае используются основные объекты *нумераторы*, которые представляют собой обычные счетчики значений.

Документы могут выступать регистраторами для регистров. В этом случае при сохранении документов используется специальная процедура *проведения*, которая обеспечивает изменение состояния связанного регистра. Таким образом, документы обеспечивают сохранение данных в постоянных хранилищах – *регистрах.*

Для облегчения ввода данных в документах могут использоваться данные справочников.

При создании документов автоматически создаются связанные с ними программные объекты.

Объект ДокументыМенеджер представляет собой коллекцию документов. Программный доступ осуществляется через экземпляр объекта по имени Документы.

Объект ДокументМенеджер является элементом коллекции и представляет отдельный документ. С помощью этого объекта осуществляется поиск документов, получение выборки документов, создание нового документа, работа с формами и макетами документа. Программный доступ осуществляется так:

Документы.Документ1.

Объект ДокументВыборка обеспечивает доступ к данным документа. Обеспечивает динамическую подгрузку данных. Программный доступ осуществляется так:

Выборка = Документы.Документ1.Выбрать();

Объект ДокументСсылка обеспечивает доступ на чтение к экземпляру документа.

Объект ДокументОбъект обеспечивает доступ на чтение и запись к экземпляру документа.

Из одних объектов с помощью определенных свойств или методов можно получить другие объекты, как показано рис. 3.

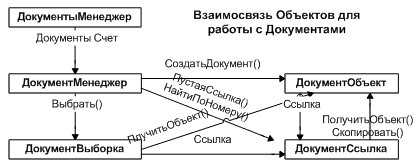


Рис.3 Взаимосвязь объектов для работы с документами

# Заключение

Объектная модель работы с данными представляет собой мощный инструмент для организации и управления информацией в различных компьютерных системах. Ее преимущества включают модульность, повторное использование кода, удобство в поддержке и расширении систем, а также более наглядное представление данных. Объектно-ориентированный подход позволяет создавать более гибкие и эффективные программные решения, а также обеспечивает более простую интеграцию с другими системами.

Однако, как и любой подход, объектная модель работы с данными имеет свои ограничения и недостатки, и ее применение требует внимательного анализа конкретной ситуации и задач. Несмотря на это, объектная модель является важной составляющей современной информационной технологии и продолжает активно развиваться с появлением новых технологий и методологий.

В целом, изучение объектной модели работы с данными позволяет лучше понять принципы организации информации в компьютерных системах и применять их в практических задачах разработки программного обеспечения.

# Список источников

1. Статья «Объектная модель системы» [Электронный ресурс], URL: <https://old.mista.ru/oop_book/glava2_1.htm>
2. Статья «Как мы в 1С: Предприятии работаем с объектной моделью данных» [Электронный ресурс], URL: <https://wonderland.v8.1c.ru/blog/kak-my-v-1s-predpriyatii-rabotaem-s-modelyami-dannykh-ili-pochemu-my-ne-rabotaem-s-tablitsami-/?sphrase_id=355816>
3. Статья «Понятное объяснение объектной модели» [Электронный ресурс], URL: <https://nauchniestati.ru/spravka/obektnaya-model-i-obektno-relyaczionnaya-proekcziya/#%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5>
4. Статья «Объектные модели данных и их реализация» [Электронный ресурс], URL: <https://www.itweek.ru/themes/detail.php?ID=47413>