

Министерство образования Республики Беларусь
БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

Кафедра «Программное обеспечение вычислительной
техники и автоматизированных систем»

МОДЕЛИРОВАНИЕ К РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В VISIO

Методические указания
для студентов специальности
1-40 01 01 «Программное обеспечение
информационных технологий»

Учебное электронное издание

Минск 2010

УДК 004.45 (075.8)

С о с т а в и т е л ь :

О.В. Бугай

Р е ц е н з е н т ы :

В.А. Кочуров, доцент кафедры «Системы автоматизированного проектирования» БНТУ, кандидат технических наук;

А.Т. Кулаков, доцент кафедры «Робототехнические системы» БНТУ, кандидат технических наук.

В методических указаниях описаны особенности моделирования программного обеспечения с использованием графического редактора VISIO.

Указания могут быть полезны студентам специальностей, связанных с программированием, и лицам, которые занимаются моделированием программного обеспечения.

Белорусский национальный технический университет
пр-т Независимости, 65, г. Минск, Республика Беларусь
Тел.(017) 293-91-97 факс (017) 292-91-37
Регистрационный № БНТУ/ФИТР49-12.2010

© Бугай О.В., 2010

© БНТУ, 2010

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1. ОПЕРАЦИИ МОДЕЛИРОВАНИЯ ПРОЦЕССОВ В VISIO ..	5
1.1. Операции построения IDEF0-моделей процессов	5
1.2. Операции построения DFD-моделей процессов	10
1.3. Операции построения IDEF3-моделей процессов	12
2. ОПЕРАЦИИ МОДЕЛИРОВАНИЯ ДАННЫХ	13
3. ОПЕРАЦИИ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО МОДЕЛИРОВАНИЯ ПО В VISIO	30
3.1. Понятие языка UML.....	30
3.2. Операции создания модели вариантов использования	30
3.3. Операции построения диаграммы состояний.....	34
3.4. Операции построения диаграммы деятельности	40
3.5. Операции построения диаграммы последовательности ..	42
3.6. Операции построения диаграммы кооперации	48
3.7. Операции построения диаграмм классов.....	51
3.8. Операции построения диаграммы компонентов.....	58
СПИСОК ЛИТЕРАТУРЫ	63

ВВЕДЕНИЕ

В качестве инструментария при моделировании процессов обычно используют среду AllFusion Process Modeler, данных – AllFusion ERwin Data Modeler, программного обеспечения – Rational Rose. Однако вместо перечисленных CASE-средств можно успешно использовать графический редактор Visio, который, не являясь CASE-средством, позволяет удовлетворять запросы не только инженеров-механиков, инженеров-электриков, экономистов, хозяйственников, но и разработчиков программного обеспечения. Используя Visio, можно строить модели программ данных, выполнять операции прямого и обратного инжинеринга. Словом, при отсутствии выше упомянутых CASE-средств, можно применять Visio.

Как показала практика применения этого редактора, наибольшие затруднения у разработчиков программ возникают из-за отсутствия литературы с кратким описанием операций, связанных с построением различных видов диаграмм, используемых при создании ПО. В предлагаемых методических указаниях предпринята попытка дать сжатое описание операций построения моделей процессов, данных и программного обеспечения в редакторе Visio.

Для предельно краткого изложения особенностей выполнения конкретных операций приняты следующие условные обозначения:

- Имя исходного пункта – Окно Browser:, Menu:, Панель Diagram и др.;
- RClick – нажатие правой кнопки мыши;
- DbClick – двойной щелчок мышью;
- → выбрать;
- = присвоить значение;
- ← = присвоить принятому значению по уполномочию новое значение;
- | ИЛИ;

- ! И;
- [Имя кнопки] – нажать кнопку с заданным именем;
- LClick_∨ – при нажатой левой кнопки мыши;
- LClick_∧ – отпустить левую кнопку мыши;
- {var 1 | var 2 | ...} – альтернатива;
- 1{ }* – повторить от 1-го до необходимого количества раз;
- () – необязательный элемент (параметр).

1. ОПЕРАЦИИ МОДЕЛИРОВАНИЯ ПРОЦЕССОВ В VISIO

1.1. Операции построения IDEF0-моделей процессов

Полагают, что нотация IDEF0 является основной. Она предназначена для описания существующих бизнес-процессов на предприятии (модель AS-IS) и идеального положения вещей – того, к чему следует стремиться (модель TO-BE). Если в процессе моделирования нужно осветить специфические стороны технологического процесса предприятия, возможно переключение на любой ветви модели на нотацию IDEF3 или DFD и создание смешанной модели.

Процесс моделирования системы в IDEF0 начинается с определения наиболее абстрактного уровня его описания (контекста). В контекст входят определение моделируемой системы, цели и точки зрения на модель.

Приступая к моделированию системы, следует определить область (Scope) моделирования. При этом необходимо учитывать два компонента – широту (границы модели) и глубину (уровень детализации завершенной модели).

Цель моделирования (Purpose) должна дать ответы на следующие вопросы:

1. Почему этот процесс должен быть смоделирован?
2. Что должна показывать модель?
3. Что может получить читатель?

Точка зрения (View Point) должна соответствовать цели моделирования и не должна изменяться в процессе моделирования. Как правило, выбирается точка зрения человека (например, руководителя предприятия), ответственного за моделируемую работу в целом. Допускается задокументировать и альтернативные точки зрения на модели. Для этой цели имеется возможность исследовать диаграммы FEO (For Exposition Only).

Технология проектирования ИС подразумевает сначала создание модели AS-IS, ее анализ и изучение бизнес-процессов, т.е. создание модели TO-BE, на основе которой строятся модель данных, прототип и затем окончательный вариант ИС.

Операции, выполняемые в нотации IDEF0, и рекомендуемая последовательность их выполнения даны в табл. 1.1.

Таблица 1.1

Операции, выполняемые в нотации IDEF0

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание новой схемы модели	Меню: Файл → Создать → Блок схема → Схема IDEF0
2. Настройка внешнего вида схемы	Меню: Фигура → Настройка макета... → <выбрать параметры {Размещение Соединительные линии}> → [OK]

<i>1</i>	<i>2</i>
3. Добавление процесса	<p>Фигуры схемы IDEF0: <Фигура «Блок действия»> → LClick_v → <Страница со схемой> → LClick_^ → <Задать имя процесса, его идентификатор, (идентификатор подчиненной схемы)> → [OK]</p>
4. Добавление элементов входа, выхода, ресурсов, управления	<p>Фигуры схемы IDEF0: <Фигура «Односторонняя соединительная линия»> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка с наконечником фигуры «Односторонняя соединительная линия»> → LClick_v → <Точка соединения фигуры «Блок действия»> → LClick_^ → DbClick → <Ввести соответствующий текст></p>
5. Добавление подписи	<p>Фигуры схемы IDEF0: <Фигура «Подпись»> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка с наконечником фигуры «Подпись»> → LClick_v → <Любая из соединительных линий процессов> → LClick_^ → DbClick → <Ввести соответствующий текст></p>

<i>1</i>	<i>2</i>
<p>6. Добавление соединяющихся линий</p>	<p>Фигуры схемы IDEF0: <Фигура «Соединительная линия IDEF0» #1> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка без наконечника фигуры «Соединительная линия IDEF0»> → LClick_v → <Точка соединения фигуры «Блок действия»> → LClick_^ → <Конечная точка с наконечником фигуры «Соединительная линия IDEF0»> → LClick_v → <Точка соединения фигуры «Блок действия»> → !{<Фигура «Соединительная линия IDEF0»> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка без наконечника фигуры «Соединительная линия IDEF0»> → LClick_v → <Точка соединения фигуры «Блок действия»> → LClick_^ → <Конечная точка с наконечником фигуры «Соединительная линия IDEF0»> → LClick_v → <Конечная точка с наконечником фигуры «Соединительная линия IDEF0» #1>}</p>

<i>1</i>	<i>2</i>
<p>7. Добавление разъединяющихся стрелок</p>	<p>Фигуры схемы IDEF0: <Фигура «Соединительная линия IDEF0» #1> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка без наконечника фигуры «Соединительная линия IDEF0»> → LClick_v → <Точка соединения фигуры «Блок действия»> → LClick_^ → <Конечная точка с наконечником фигуры «Соединительная линия IDEF0»> → LClick_v → <Точка соединения фигуры «Блок действия»> → !{<Фигура «Соединительная линия IDEF0»> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка без наконечника фигуры «Соединительная линия IDEF0»> → LClick_v → <Совместить с конечной точкой без наконечника первоначальной соединительной линии> → LClick_^ → <Конечная точка с наконечником фигуры «Соединительная линия IDEF0»> → LClick_v → <Точка соединения фигуры «Блок действия»>}></p>

<i>1</i>	<i>2</i>
8. Туннелирование стрелок	<Фигура «Соединительная линия IDEF0»> → (<Конечная точка без наконечника фигуры «Соединительная линия IDEF0»> <Конечная точка с наконечником фигуры «Соединительная линия IDEF0»>) → RClick → <Выбор опции (Туннелировать вход Туннелировать выход)>

1.2. Операции построения DFD-моделей процессов

Обычно нотация DFD является дополнением к IDEF0. Как уже упоминалось выше, переключение с нотации IDEF0 на DFD возможно на любой ветви модели. Это позволяет более полно показать моделируемый процесс. В Visio для построения DF-диаграмм можно использовать как нотацию Гейна–Сарсона, так и нотацию Йодана. Разумеется, можно начинать моделирование процессов сразу и с нотации DFD, трактуя ее как базовую.

Ниже в табл. 1.2 приведены операции и последовательность их выполнения при построении DFD-моделей процессов.

Таблица 1.2

Операции, выполняемые при построении DF-диаграмм

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание новой схемы модели	Меню: Файл → Создать → Блок схема → Схема потоков данных
2. Настройка внешнего вида схемы	Меню: Фигура → Настройка макета... → <выбрать параметры {Размещение Соединительные линии}> → [OK]
3. Добавление состояния	Фигуры схемы DFD: <Фигура «Состояние»> → LClick_v → <Страница со схемой> → LClick_^ → DbfClick → <Ввести соответствующий текст>
4. Добавление сущности	Фигуры схемы DFD: <Фигура «Сущность»> → LClick_v → <Страница со схемой> → LClick_^ → DbfClick → <Задать имя сущности>
5. Добавление хранилища данных	Фигуры схемы DFD: <Фигура «Хранилище данных»> → LClick_v → <Страница со схемой> → LClick_^ → DbfClick → <Определить название базы данных>

1	2
6. Добавление связи между состояниями	<p>Фигуры схемы DFD: <Фигура «Центр-к-центру»> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка без наконечника фигуры «Центр-к-центру»> → LClick_v → <Центр фигуры «Состояние»> → LClick_^ → <Конечная точка с наконечником фигуры «Центр-к-центру»> → LClick_v → <Центр фигуры «Состояние»> → LClick_^ → 2 {<Желтый ромбик с подсказкой «Изменить размер»> → LClick_v → <Точка соединения фигуры «Состояние»> → LClick_^} → Db1Click → <Определить название связи></p>

1.3. Операции построения IDEF3-моделей процессов

Обычно нотация IDEF3 является дополнением к IDEF0 либо DFD. При необходимости можно начинать моделирование процессов сразу и с нотации IDEF3.

К сожалению, в Visio отсутствует трафарет для построения диаграмм IDEF3, но это не значит, что в этом редакторе невозможно реализовать их построение. При необходимости пользователь может построить шаблоны всех элементов, используемых при этом, и создать новый трафарет, наполненный этими элементами.

2. ОПЕРАЦИИ МОДЕЛИРОВАНИЯ ДАННЫХ

Visio позволяет представить модель данных как на физическом, так и на логическом уровнях. На логическом уровне данные не связаны с конкретной системой управления базой данных (СУБД) и могут быть проиллюстрированы даже для неспециалистов. Физический уровень данных – это по сути отображение системного каталога, зависящего от реализации СУБД.

Операции к логическому и физическому моделированию данных, а также последовательность их выполнения представлены в табл. 2.1.

Таблица 2.1

Операции построения логической и физической модели данных

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание новой схемы модели с нуля	Меню: Файл → Создать → Программное обеспечение... → Схема модели Базы данных → База данных → Параметры... → Документ... → <выбрать параметры {Общие Таблица Отношение}> → [ОК]

<i>1</i>	<i>2</i>
<p>2. Использование существующей базы данных в качестве основы</p>	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели Базы данных → База данных → Реконструировать → <выбрать {драйвер базы данных Microsoft Office Visio (СУБД) Настройка 1* драйвер ODBC 2*}> → <выбрать (источник данных обновляемой БД) [Создать] 3*> → <выбрать требуемые параметры> → [Далее] → <следовать инструкциям в диалоговых окнах> → <установить флажки для типов сведений, которые требуется извлечь> → [Далее] → <установить флажки для таблиц, которые требуется извлечь [Выделить все]> → [Далее] → <выбрать процедуры, которые требуется извлечь [Выделить все]> → [Далее] → <выбрать, будет ли автоматическое добавление элементов> → <просмотреть выбранные параметры> → [Готово]</p> <p>Примечания: 1* - если драйвер еще не связан источником данных; 2* - если лист Excel; 3* - если еще не создан источник данных.</p>

1	2
<p>3. Импорт и уточнение существующей модели</p>	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели Базы данных → База данных → Импорт → Импорт <тип_модели> → <(ввести путь и имя файла для модели [Обзор] → <выбрать файл> → [Открыть])> → [ОК] → окно "Таблицы и представления" → <выбрать и перетащить таблицы для моделирования на страницу документа></p> <p>Примечание: только для файлов моделей VisioModeler версии 2.0 или более поздних (IMD) и для файлов моделей PLATINUM ERwin версий 2.6, 3.0 и 3.52 (ERX).</p>
<p>4. Создание таблицы в схеме модели базы данных</p>	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели Базы данных → Отношение сущности → <фигура "Сущность"> → LClick_v → <Страница документа> → LClick_^ → <фигура "Сущность"> → DblClick → Категории → Определение = → <имя таблицы> → Столбцы = → 1 {<имя данных ! тип данных> → <требуемые столбцы> → <установить флажок "Обязательное"> → <определяющая строка таблицы> → <установить флажок "PK">}* → {Индексы Триггеры Проверка Дополнительные} → <требуемые настройки> → 1 {[ОК]}*</p>

<i>1</i>	<i>2</i>
5. Добавление и изменение свойств для столбцов	<p><Таблица>: DblClick → Категории → Столбцы → <первая пустая ячейка “Физическое имя”> = → <Ввести имя> → Тип данных → <выбрать тип данных ввести тип данных в список> → <установить флажок “Обязательное”> → п.6 → [ОК]</p>
6. Установка первичного ключа	<p><Таблица>: DblClick → Категории → Столбцы → <определяющая строка таблицы> → <установить флажок ”PK”> → [ОК]</p>
7. Удаление таблицы	<p>Схема: <требуемая таблица> → <клавиша DEL> → диалоговое окно “Удалить объект” → {[Да] 1* [НЕТ] 2*} → База данных → Параметры → Моделирование → Логическая схема → <выбрать нужный параметр></p> <p>Примечания: 1* - чтобы удалить таблицу из документа и основной модели; 2* - чтобы удалить таблицу только из документа.</p>

<i>1</i>	<i>2</i>
<p>8. Создание связи между таблицами</p>	<p>Страница документа: <таблица, которая будет на стороне первичного ключа в связи> → DbfClick → Категории → Столбцы → <выделить столбец для однозначного определения каждой строки таблицы> → <установить флажок РК> → [OK] → Отношение сущности → <фигура ”Отношение”> → LClick_v → <Страница документа> → LClick_^ → <верхний конец фигуры> → LClick_v → <родительская таблица> → LClick_^ → <другой конец фигуры> → LClick_v → <дочерняя таблица> → LClick_^</p> <p>Примечание: если линии связи отсутствуют – Меню → База данных → Параметры → Документ → Отношения → Показывать → <установить флажок ”Отношения”></p>

1	2
<p>9. Удаление связи из модели базы данных</p>	<p><Требуемая связь>: <клавиша DEL> → диалоговое окно Удалить объект → {[Да] 1* [НЕТ] 2*} → (База данных → (Показать связанные таблицы 3* Параметры → Моделирование → Логическая схема → <выбрать требуемый вариант>)</p> <p>Примечания: 1* - чтобы удалить связь из документа и основной модели; 2* - чтобы удалить только линию в документе; 3* - если была нажата [НЕТ], а позднее было решено добавить линию назад в документ.</p>
<p>10. Определение связи "один-к-одному" или "один-к-многим"</p>	<p><Требуемая связь>: Db1Click → Категории → Прочее → Мощность → <выбрать размерность, наиболее подходящую к связи></p> <p>Примечание: для связи «один-к-многим» наилучшим вариантом будет 0 или более или 1 или более; для отношений «один-к-одному» — 0 или 1 или Ровно 1.</p>
<p>11. Отображение связей и их свойств в схеме</p>	<p>Меню: База данных → Параметры → Документ → Отношение → <{отобразить скрыть (Связи Размерность Нотацию "гусиные лапки" Глагольную фразу Ссылочные действия)} ></p>

1	2
<p>12. Установка действий для обеспечения целостности данных</p>	<p><Связь, для которой требуется установить действие >: DbClick → Категории → Действие ссылочной целостности → <выбрать действие {Без действия Каскад Присвоить пустое значение По умолчанию Не обеспечивать}> Примечание: если параметры ”Присвоить пустое значение” и “По умолчанию” недоступны, вероятно, для связи выбран параметр “Необязательный” на вкладке Прочее окна “Свойства базы данных”.</p>
<p>13. Задание значения по умолчанию для столбца</p>	<p><Требуема таблица>: DbClick → Категории → Столбцы → <требуемый столбец> → [Изменить] → Определение → <ввести значение по умолчанию></p>
<p>14. Отображение целостности данных</p>	<p>Меню: База данных → Параметры → Документ → Отношение → Показывать → <установить флажок ”Действия ссылочной целостности”></p>
<p>15. Изменение размерности связи</p>	<p>Связь: DbClick → Категории → Прочее → Мощность → <выбрать размерность> → Click <в любом месте документа ></p>

1	2
16. Создание индексов	<p>Схема модели базы данных: <требуемая таблица> → DblClick → Категории → Индексы → [Создать] ⇒ <имя для индекса> → [OK] → Тип индекса → <выбрать уникальный неуникальный> → Доступные столбцы → <выбрать имя каждого столбца, который требуется включить в этот индекс> → [Добавить] → Индексированные столбцы → <установите снять флажок "Возрастание"> → Click <в любом месте документа ></p>
17. Удаление индекса	<p>Схема модели базы данных: <требуемая таблица> → DblClick → Категории → Индексы → Имя индекса → <имя индекса, который требуется удалить> → [Удалить]</p>
18. Изменение свойств индекса	<p>Схема модели базы данных: <требуемая таблица> → DblClick → Категории → Индексы → Имя индекса → <имя индекса, который требуется изменить> → ([Переименовать] 1* Тип индекса → <нужный параметр> 2* [Параметры] 3*)</p> <p>Примечания: 1* - чтобы указать новое имя для индекса; 2* - чтобы указать (не)уникальный индекс; 3* - чтобы указать атрибуты индекса, зависящие от драйвера.</p>

<i>1</i>	<i>2</i>
<p>19. Добавление, удаление и изменение порядка столбцов в индексе</p>	<p>Схема модели базы данных: <требуемая таблица> → DbClick → Категории → Индексы → Имя индекса → <имя индекса, который требуется изменить> → (Доступные столбцы → <выбрать имя требуемого столбца> → [Добавить] 1* Индексированные столбцы → <выбрать имя требуемого столбца> → [Удалить] 2* Индексированные столбцы → <выбрать имя требуемого столбца> → LClick_v → <новое место в списке> → LClick_^ 3*)</p> <p>Примечания: 1* - чтобы добавить столбец к индексу; 2* - чтобы удалить столбец из индекса; 3* - чтобы изменить порядок столбцов.</p>
<p>20. Добавление представления в модель базы данных</p>	<p>{Отношение сущности Объекты реляционной базы данных}: <фигура "Представление"> → LClick_v → <Страница документа> → LClick_^ → Click <в любом месте документа ></p>

Продолжение табл. 2.1

<i>1</i>	<i>2</i>
<p>21. Назначение представлению столбцов из таблиц</p>	<p><Требуемое Представление>: Db1Click → Категории → Столбцы → (Физическое имя = → <ввести имя столбца> [Добавить] → <имя столбца по умолчанию> → <ввести новое имя>) → 1 {<имя столбца> → [Изменить] → Источник → На известном столбце в других таблице или представлении → [Изменить] → Выбор столбца → <выбрать столбец из списка таблиц и представлений> → 1 {[ОК]}*}*</p>
<p>22. Изменение свойств столбца в представлении</p>	<p><Требуемое Представление>: Db1Click → Категории → Столбцы → <столбец, который требуется изменить> → [Изменить] → <требуемая вкладка> → <внести изменения> → [ОК]</p>
<p>23. Изменение свойств представления</p>	<p><Требуемое Представление>: Db1Click → Категория → <свойство, которое требуется изменить> → <внести требуемые изменения></p>

1	2
<p>24 .Удаление представления</p>	<p>Схема: <требуемое представление> → <клавиша DEL> → диалоговое окно “Удалить объект” → {[Да] 1* [НЕТ] 2*} → (База данных → Параметры → Моделирование → Логическая схема → При удалении объекта из схемы → Спрашивать пользователя Таблицы и представления → <выбрать требуемое представление> → <клавиша DEL> 3*)</p> <p>Примечания: 1* - чтобы удалить представление из модели и окна Таблицы и представления; 2* - чтобы удалить представление только со страницы документа; 3* - чтобы удалить представление из модели.</p>
<p>25. Изменение порядка представлений в окне ”Код”</p>	<p>Меню: База данных → Представление → Код → <представление> → LClick_v → <новое место в окне> → LClick ^</p>
<p>26. Определение пространства имен для сущности или представления</p>	<p><Требуемая {Сущность Представление}>: DblClick → Категории → Определение → Пространство имен = → <ввести описание></p>

1	2
<p>27. Создание текста программы для представлений</p>	<p>{Отношение сущности Объекты реляционной базы данных}:
 <фигура "Представление"> → LClick_v → <Страница документа> → LClick_^ → DbfClick → Категории → SQL → <ввести текст программы></p>
<p>28. Определение категории в схеме модели базы данных</p>	<p>{Отношение сущности Объекты реляционной базы данных}:
 <фигура "Категория"> → LClick_v → <Страница документа> → LClick_^ → <фигура "Родительская таблица и категория"> → LClick_v → <Страница документа> → LClick_^ → <один конец фигуры "Родительская таблица и категория"> → LClick_v → <центр родительской таблицы> → LClick_^ → <другой конец фигуры "Родительская таблица и категория"> → LClick_v → <центр фигуры "Категория"> → LClick_^ → 1 {<фигура "Категория и дочерняя таблица"> → LClick_v → <Страница документа> → LClick_^ → <один конец фигуры "Категория и дочерняя таблица"> → LClick_v → <центр фигуры "Категория"> → LClick_^ → <другой конец фигуры "Категория и дочерняя таблица"> → LClick_v → <центр дочерней таблицы> → LClick_^}*
</p>

1	2
<p>29. Определение категории с помощью инструмента "Соединительная линия"</p>	<p>Схема модели базы данных: {Отношение сущности Объекты реляционной базы данных} → <фигура "Категория"> → LClick_v → <Страница со схемой> → LClick_^ → панель инструментов Стандартная → [Соединительная линия] → <центр родительской таблицы> → LClick_v → <центр фигуры "Категория"> → LClick_^ → 1 {< центр фигуры "Категория"> → LClick_v → <центр дочерней таблицы> → LClick_^}*</p>
<p>30. Установка фигуры "Категории" как завершенной или незавершенной</p>	<p><Схема модели базы данных>: <фигура "Категория"> → DblClick → Категории → Определение → <(снять установить) флажок "Категория завершенна"></p>
<p>31. Определение атрибута определителя категории</p>	<p><Схема модели базы данных>: <фигура "Категория"> → DblClick → Категории → Определение → Данный атрибут → <столбец из родительской таблицы, который требуется задать как определитель категории></p>

1	2
32. Отображение типов данных в схеме модели базы данных	Меню: База данных → Параметры → Документ → Таблица → Типы данных → {Показывать физические Показывать переносимые} → ОК
33. Задание дополнительных свойств для таблиц и представлений	< Требуемая {таблица представление} >: DblClick → Категории → Дополнительные → <настроить свойства> → Click <в любом месте документа >
34. Создание правил контроля для столбца	Таблица: DblClick → Категории → Столбцы → <требуемый столбец> → [Изменить] → Проверка → <ввести требуемые ограничения> → Click <в любом месте документа >
35. Создание правил контроля для таблицы	Таблица: DblClick → Категории → Проверка → [Добавить] → Свойства → <ввести имя для правила контроля> → Основной текст → <ввести текст программы> → [ОК]
36. Создание хранимых процедур и пользовательских функций	Меню: База данных → Представление → Код → Глобальный код → [Создать] → Свойства → <выбрать вид текста программы> = → <имя> → Основной текст → <ввести текст программы> → [ОК]

<i>1</i>	<i>2</i>
37. Создание триггеров	<p><Требуемая Таблица>: DbClick → Категории → Триггеры → [Добавить] → Свойства = → <имя триггера> → Основной текст → <ввести текст программы> → [ОК]</p>
38. Отображение сведений о таблице	<p>Меню: База данных → Параметры → Документ → Таблицы → <выбрать элементы, которые требуется отобразить в схеме></p> <p>Примечание: в схеме можно отобразить сведения о таблицах – первичный ключ (PK), внешние ключи (FKn), альтернативные ключи (AKn), уникальность (UN), индексы (In) и типы данных.</p>
39. Создание связи с внешним ключом с помощью средства ”Соединительная линия”	<p>Схема модели базы данных: панель инструментов Стандартная → [Соединительная линия] → <центр родительской таблицы> → LClick_v → <центр дочерней таблицы> → LClick_^ → <отношение> → DbClick → <категория с данными для изменения> → <внести необходимые изменения></p>

1	2
<p>40. Выбор нотации размерности отношений: IDEF1X, реляционная или «птичья лапка»</p>	<p>Меню: База данных → Параметры → Документ → Общие → Набор символов → (IDEF1X Реляционная) → Отношение → Показывать → <установить флажок "Птичья лапка"> → [ОК]</p>
<p>41. Изменение набора доступных физических типов данных</p>	<p>Меню: База данных → Параметры → Драйверы → <драйвер Visio для целевой СУБД> → [Настройка] → <установить флажок для соответствующего драйвера ODBC></p>
<p>42. Создание пользовательского типа данных</p>	<p>Меню: База данных → Пользовательские типы → [Добавить] = → <имя нового пользовательского типа> → <установите флажок "Копировать из"> → <выбрать имя существующего пользовательского типа данных> → [ОК] → <указать характеристики переносимого типа данных для этого типа данных> → [ОК]</p>
<p>43. Установка типа набора для столбца</p>	<p><Таблица со столбцом, которому требуется задать тип набора: DbClick → Категории → Столбцы → <столбец, который требуется задать как тип набора> → [Изменить] → Коллекция → {Множество Список Мультимножество} → [ОК]</p>

<i>1</i>	<i>2</i>
44. Создание составного типа данных с помощью фигуры «Тип»	<p>Объекты реляционной базы данных: <фигура "Тип"> → LClick_v → <Схема модели базы данных> → LClick_^ → <фигура "Тип"> → DbClick → Категории → Поля → <пустая строка> → <ввести новое Физическое имя> → <указать тип данных> → <установите флажок в столбце Обязательное> → Определение → <определить значение Тип> → <выбрать требуемые параметры {Именованный тип-строка Индивидуальный тип Домен Тип-семейство псевдонима}></p>

3. ОПЕРАЦИИ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО МОДЕЛИРОВАНИЯ ПО В VISIO

3.1. Понятие языка UML

Что касается нотации, то следует отдать предпочтение унифицированному языку моделирования UML (Unified Modeling Language).

Создание UML началось в 1994 г., когда Буч и Рамбо приняли попытку объединить свои методы ООА и проектирования ПО. В 1995 г. к ним присоединился Якобсон. Первая версия UML появилась в 1997 г., а версия UML 1.1 – в конце 1997 г. и была утверждена и принята на вооружение всеми крупнейшими производителями ПО.

Создатели UML представляют его как язык определения, представления, проектирования и документирования программных систем различной природы.

3.2. Операции создания модели вариантов использования

Visio весьма удобный инструмент для построения диаграмм вариантов использования. Кроме того в Visio при построении упомянутых диаграмм имеется возможность заключить область предмета разработки в рамку. Это, на наш взгляд, улучшает внешний вид диаграммы.

Операции к построению диаграмм вариантов использования (диаграмм прецедентов) и последовательность их выполнения даны в табл. 3.1.

Таблица 3.1

Операции построения диаграмм вариантов использования

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание схемы сценариев выполнения	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели UML → Сценарий выполнения UML → <фигура “Граница системы”> → LClick_v → <Страница документа> → LClick_^ → 1 {Сценарий выполнения UML → 1 {<фигура “Сценарий выполнения”> → LClick_v → <Разместить внутри Границы системы> → LClick_^ }* → 1 {<фигура “Актер”> → LClick_v → <Разместить вне Границы системы> → LClick_^ }* → 1 {<фигура “Сообщение” “Расширение” “Использование”> → LClick_v → < Страница документа > → LClick_^ }* }</p>

1	2
<p>2. Обозначение границы системы</p>	<p>Окно Документа: Сценарий выполнения UML → <фигура “Граница системы”> → LClick_v → <Страница документа> → LClick_^ → DbClick ==> <Новое имя системы> [Delete] → Click <За пределами фигуры> → (<фигура “Граница системы”> → <Маркер выделения {верхний нижний левый правый}> → LClick_v → {вверх вниз влево вправо} → LClick ^)</p>
<p>3. Обозначение отношения между актером и сценарием выполнения</p>	<p>Окно Документа: Сценарий выполнения UML → <фигура “Сообщение”> → LClick_v → <Страница документа> → LClick_^ → <Конечная точка фигуры “Сообщение”> → LClick_v → <Точка соединения фигуры “Актер”> → LClick_^ → <Вторая конечная точка фигуры “Сообщение”> → LClick_v → <Точка соединения фигуры “Сценарий выполнения”> → LClick ^</p>

1	2
<p>4. Обозначение направления информационного потока</p>	<p><Фигура “Сообщение”> → DbfClick → Окончания ассоциаций → <Выделить имя окончания> → [Свойства...] → <флажок IsNavigable (другие свойства)> → Click → [Ok] → [Ok] → <Фигура “Сообщение”> → Параметры отображения фигуры → Параметры окончаний → Перемещаемости навигаций → Click (установить флажок, если нет по умолчанию) → [Ok]</p>
<p>5. Отображение отношений “Использование” между двумя сценариями выполнения</p>	<p>Окно Документа: Сценарий выполнения UML → <фигура “Использование”> → LClick_v → <Страница документа> → LClick_^ → <Конечная точка без наконечника фигуры “Использование”> → LClick_v → <Точка соединения фигуры “Сценарий выполнения”, которая использует поведение другого сценария выполнения> → LClick_^ → <Конечная точка с наконечником фигуры “Использование”> → LClick_v → <Точка соединения используемой фигуры “Сценарий выполнения”> → LClick_^ → <фигура “Использование”> → DbfClick → Свойства обобщения UML → <Добавить требуемые значения свойств> → [Ok]</p>

1	2
<p>6. Отображение отношений “Расширения” между двумя сценариями выполнения</p>	<p>Окно Документа: Сценарий выполнения UML → <фигура Расширение> → LClick_v → <Страница документа> → LClick_^ → <Конечная точка без наконечника фигуры “Расширение”> → LClick_v → <Точка соединения фигуры “Сценарий выполнения”, обеспечивающая расширение> → LClick_^ → <Конечная точка с наконечником фигуры “Расширение”> → LClick_v → <Точка соединения основной фигуры “Сценария выполнения”> → LClick_^ → <фигура “Расширение”> → DbClick → Свойства обобщения UML → <Добавить требуемые значения свойств > → [Ok]</p>

3.3. Операции построения диаграммы состояний

В Visio весьма прост процесс построения диаграмм состояний. Операции к построению диаграмм состояний и последовательность выполнения каждой из них даны в табл. 3.2.

Операции построения диаграммы состояний

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание схемы состояния	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели UML... → 1 {Схема состояний UML → 1 {<фигура “Состояние”> → LClick_v → <Страница документа> → LClick_^}* → 1 {<фигура “Составное состояние”> → LClick_v → <Страница документа> → LClick_^}* → 1 {<фигура “Переход”> → LClick_v → <Страница документа> → LClick_^ → <Конечная точка без наконечника фигуры “Переход”> → LClick_v → <Точка соединения фигуры “СостояниеN”> → LClick_^ → <Конечная точка с наконечником фигуры “Переход”> → LClick_v → <Точка соединения фигуры “СостояниеN+1”> → LClick_^}* → 1 {<фигура “Переход”> → LClick_v → <фигура “Переход” (объединение) “Переход” (разветвление)> → LClick_^}*}*}</p> <p>Примечание: DbClick по фигуре → окно Свойства UML → указать имя, действия, деятельность, события и др. свойства.</p>

1	2
<p>2. Добавление внутреннего действия или деятельности в состояние</p>	<p><Страница со схемой>: <фигура “Состояние”> → DbClick → <Окно Свойства состояния UML> → Внутренние переходы → =<Ввести имя перехода> → [Свойства] → <Выбрать событие 1*> → =<Ввести имя события> → [Ok] → <Установить флажок Условие> → <Тело> → =<Ввести текст> → <Язык> → <Выбрать язык> → Действия → [Создать] → <Выбрать действие> → [Ok] → [Свойства] → =<Ввести имя действия> → (Сведения Ограничения Именованные значения → =<Ввести требуемые сведения>) → 1 {[Ok]}* → <фигура ”Состояние”> → RClick → Параметры отображения фигуры → Скрыть → <Снять флажок Переход> Примечание 1*: [События] → [Создать] → <Выбрать вид события> → [Ok]</p>

1	2
<p>3. Добавление действий при входе и при выходе</p>	<p><Страница со схемой>: <фигура “Состояние”> → DbClick → <Окно Свойства состояния UML> → Вход Выход → [Создать...] → <Выбрать тип действия> → [Ok] → =<Ввести имя действия> → [Свойства] → <Ввести (сведения ограничения именованные значения)> → 1 {[Ok]}* → <фигура “Состояние”> → RClick → Параметры отображения фигуры → Скрыть → <Снять флажок Переход></p>
<p>4. Создание составного или вложенного состояния</p>	<p>Вкладка Схема состояний UML: <фигура “Составное состояние”> → LClick_v → <Страница со схемой> → LClick_^ → (создалась новая страница документа) Click <Новая страница> → 1 {Схема состояний UML → <фигура “Состояние” “Переход” “Историческое состояние” “Состояние глубокой истории” и т.д.> → LClick_v → <Страница> → LClick_^ }* - отобразить параллельные, последовательные или вложенные подсостояния данного составного состояния.</p>

1	2
<p>5. Добавление строки перехода к переходу</p>	<p><Фигура “Переход” “Поток управления”>: DblClick → <Окно Свойства перехода UML> → Переход → [События] → [Создать] → <Выбрать тип события> → [Ok] → =<Ввести имя события значения других свойств> → [Ok] (вернуться в категорию Переход) → <установить флажок Условие> → Текстовое поле Тело → =<Ввести условие> → Язык → <Выбрать язык> → Действия → [Создать] → <Выбрать тип действия> → [Ok] → Свойства → =<Ввести имя действия значения других свойств> → 1 {[Ok]}* - пока не закроется диалоговое окно.</p>
<p>6. Обозначение перехода между состояниями</p>	<p><Схема состояний UML>: <фигура “Переход”> → LClick_v → <Страница документа со схемой> → LClick_^ → <Конечная точка без наконечника фигуры “Переход”> → LClick_v → <Точка соединения исходной фигуры “Состояние”> → LClick_^ → <Конечная точка с наконечником фигуры “Переход”> → <Точка соединения конечной фигуры “Состояние”> → LClick_^ → <фигура “Переход”> → DblClick → <Окно Свойства перехода UML> → <Добавить строку перехода, условие, выражение действия> → [Ok]</p>

1	2
<p>7. Обозначение перехода типа “развилка”</p>	<p>< Схема состояний UML >: <фигура Переход (разветвление)> → LClick_v → <Страница документа со схемой> → LClick_^ → <Конечная точка без наконечника фигуры “Переход”> → LClick_v → <Точка соединения исходной фигуры “Состояние”> → LClick_^ → <Конечная точка с наконечником фигуры “Переход”> → LClick_v → <Точка соединения черты фигуры “Переход (разветвление)”> → LClick_^ → 1 {< Конечная точка состояния без наконечников фигуры “Переход”> → LClick_v → <Конечная точка черты фигуры “Переход” (разветвление) > → LClick_^ → <Конечная точка с наконечником фигуры “Переход”> → LClick_v → <Точка конечного состояния фигуры “Переход (разветвление)”> → LClick_v}* Примечание: с помощью фигуры “Переход” соединяется исходное состояние с чертой “Переход” (разветвление), а также черта “Переход” (разветвление) с конечными состояниями.</p>

1	2
8. Обозначение перехода типа “синхронизация”	<p><Схема состояний UML>: <фигура “Переход (объединение)”> → LClick_v → <Страница документа со схемой> → LClick_^ → 1 {<Конечная точка без наконечников фигуры “Переход”> → LClick_v → <Точка соединения исходной фигуры “Состояние”> → LClick_^ → <Конечная точка с наконечником фигуры “Переход”> → LClick_v → <Точка соединения черты фигуры “Переход” (объединение)> → LClick_^}* → <Конечная точка без наконечника фигуры “Переход”> → LClick_v → <Точка соединения черты фигуры “Переход” (объединение)> → LClick_^ → <Конечная точка с наконечником фигуры “Переход”> → LClick_v → <Точка соединения конечной фигуры “Состояние”> → LClick_^.</p>

3.4. Операции построения диаграммы деятельности

Упомянутые операции представлены в табл. 3.3, где перечислены операции, связанные с построением диаграмм деятельности, и последовательность выполнения каждой из этих операций.

Операции построения диаграммы деятельности

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание новой схемы модели	Меню: Файл → Создать → Программное обеспечение и базы данных → Схема модели UML
2. Настройка внешнего вида схемы	Меню: Фигура → Настройка макета... → <выбрать параметры {Размещение Соединительные линии}> → [OK]
3. Добавление начального, конечного состояния	Фигуры, Деятельность UML: <Фигура «Начальное состояние»> <Фигура «Конечное состояние»> → LClick_v → <Страница со схемой> → LClick_^
4. Добавление состояния действия	Фигуры, Деятельность UML: <Фигура «Состояние действия»> → LClick_v → <Страница со схемой> → LClick_^ → DbClick → <Задать имя состояния>
5. Добавление решения	Фигуры, Деятельность UML: <Фигура «Решение»> → LClick_v → <Страница со схемой> → LClick_^

1	2
6. Добавление связи между состоянием и решением	Фигуры, Деятельность UML: <Фигура «Поток управления»> → LClick_v → <Страница со схемой> → LClick_^ → <Конечная точка без наконечника фигуры «Поток управления»> → LClick_v → <Граница фигуры «Решение»> → LClick_^ → <Конечная точка с наконечником фигуры «Поток управления»> → LClick_v → <Граница фигуры «Состояние»> → LClick_^ → RClick → <Фигура «Поток управления»> → <Опция «Прямоугольная соединительная линия»>
7. Описание вида действия между состоянием и решением	(RClick → <Фигура «Поток управления»> → <Опция «Свойства»> DbClick → <Фигура «Поток управления»>) → [Действия] → [Создать] → <Выбрать тип действия из предложенных> → [OK] → <Задать название действия> → [OK]

3.5. Операции построения диаграммы последовательности

При построении диаграммы последовательности в Visio аналитик может по своему усмотрению включать/выключать измерение сообщений, включать/выключать отображение фокусов управления. Операции к построению диаграммы последовательности и последовательность выполнения каждой из них приведены в табл. 3.4.

Операции построения диаграммы последовательности

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание схемы последовательности	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели UML... → 1 {Схема последовательности UML → 1 {<фигура “Линия жизни объекта”> → LClick_v → <Страница документа> → LClick_^}* → 1 {п.4}* → <фигура Линия жизни объекта> → Db1Click → <окно Свойства роли классификатора UML > → <Роль классификатора> → {Выбрать классификатор [Создать]} → [Ok] → Схема последовательности UML → 1 {<фигура “Активация”> → LClick_v → <фигура “Линия жизни объекта”> → LClick_^ → <Фигура “Сообщение”> → LClick_v → <фигура “Линия жизни объекта”> → LClick_^ → <фигура “Сообщение”> → Db1Click → <окно Свойства UML> = → <Ввести имя и значения свойств> → [Ok] }*</p>

<i>1</i>	<i>2</i>
2. Добавление имени и значений других свойств в фигуру “Линия жизни объекта”	<Фигура “Линия жизни объекта”>: DbClick → <окно Свойства элемента UML> = → <Ввести имя> → [Ok]
3. Управление отображением свойств в фигуре “Линия жизни объекта”	<Фигура ”Линия жизни объекта”>: RClick → <Параметры отображения фигуры> → <установить снять соответствующие флажки> → [Ok]
4. Отображение маркера уничтожения	<Фигура ”Линия жизни объекта”>: RClick → Параметры отображения фигуры → <установить флажок “Маркер уничтожения”> → [Ok]
5. Применение маркера уничтожения только к выделенной фигуре	<Фигура “Линия жизни объекта”>: RClick → Параметры отображения фигуры → <установить флажок “Маркер уничтожения”> → <установить флажок “Применить к выделенным фигурам UML того же типа на текущей странице окна документа” > → [Ok]

<i>1</i>	<i>2</i>
<p>6. Указание и изменение времени начала или завершения активности объекта</p>	<p>Схема последовательности UML: 1 {<фигура “Активация”> → LClick_v → < фигура “Линия жизни объекта”> → LClick_^ → <Начальная конечная точка фигуры “Активация”> → LClick_v → <Верхние нижние точки соединения фигуры “Линия жизни объекта”> (тянуть пока точка фигуры “Активация” не совместится с требуемым временем) → LClick_^ }*</p>
<p>7. Присвоение имени активности и добавление значений других свойств</p>	<p><Фигура “Активация”>: DblClick → <окно Свойства класса UML> ⇒ <Ввести требуемые параметры> → 1 {[Ok]}*</p>

1	2
<p>8. Указание времени перехода на схеме</p>	<p><Страница документа со схемой >: панель инструментов → A → <Место на странице для сообщения, которое будет являться временем отправки сообщения> Click = → <Введите имя> → <Пустое место на странице> Click → панель инструментов → Указатель → Схема последовательности UML → <фигура “Ограничение”> → Click_v → <Страница документа> (рядом с именем сообщения) → Click_^ → <фигура “Ограничение”> → DbClick → <поле Тело> = → <Ввести параметры ограничения> → [Ok]</p> <p>Примечание: если сообщение доставляется тотчас же, то линия сообщения времени получения должна быть наклонной и иметь то же имя, что и время отправки этого сообщения только с добавлением штриха на конце.</p>

1	2
9. Отображение обусловленности в объекте схемы	<p><Страница документа со схемой >: Схема последовательности UML → <фигура “Линия жизни”> → LClick_v → <фигура “Линия жизни объекта”> → LClick_^ → {верхний нижний управляющий маркер фигуры ”Линия жизни”} → LClick_v → <Потянуть в нужную сторону> → LClick_^ → <фигура “Линия жизни”> → DbClick → <окно Свойства UML> ⇒ <Ввести имя!значения свойств> → [Ok]</p>
10. Добавление имени и значений других свойств в фигуру “Сообщение”	<p><Фигура “Сообщение”>: DbClick → <окно Свойства сообщения UML> = → <Ввести имя и требуемые параметры> → 1 {[Ok]}*</p>
11. Управление отображением свойств в фигуре “Сообщение”	<p><Фигура “Сообщение”>: RClick → Параметры отображения фигуры → {установить снять требуемые флажки} → [Ok]</p>

3.6. Операции построения диаграммы кооперации

Операции к построению диаграммы кооперации и последовательность выполнения каждой из них приведены в табл. 3.5.

Таблица 3.5

Операции построения диаграммы кооперации
(взаимодействий)

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание схемы взаимодействий	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели UML... →</p> <p>1 {Взаимодействия UML →</p> <p>1 {<фигура “Роль классификатора”> → LClick_v → <Страница документа> → LClick_^ }* →</p> <p>1 {<фигура “Несколько объектов”> → LClick_v → <Страница документа> → LClick_^ }* → 1 {п.2 7}* →</p> <p>1 {<фигура “Роль ассоциации”> → LClick_v → <Страница документа> → LClick_^ → 1 {п.9!10}*}*</p>
2. Присвоение имени фигуре «Роль классификатора» и добавление значений других свойств	<p><Фигура “Роль классификатора”>: Db1Click = → <Ввести имя! Значения свойств> → [Ok]</p>

1	2
<p>3. Управление значениями свойств, которые отображаются в фигуре «Роль классификатора»</p>	<p><Фигура “Роль классификатора”>: RClick → Параметры отображения фигуры → <установить!снять соответствующие флажки> → [Ok]</p>
<p>4. Скрытие имени того класса, с которым ассоциирована фигура «Классификатор»</p>	<p><Фигура “Роль классификатора”>: RClick → Параметры отображения фигуры → <Снять флажок “Имя классификатора” > → [Ok]</p>
<p>5. Обозначение объекта в схеме как нового, уничтоженно-го или временного</p>	<p><Фигура “Роль классификатора”>: RClick → Параметры отображения → Состояние роли классификатора → <Выбрать требуемое условие> → [Ok]</p> <p>Примечание: диалоговое окно Параметры отображения фигуры → <снять ! установить флажок Применить к выделенным фигурами UML того же типа на текущей странице окна документа ! Применить к размещаемым фигурам UML того же типа на текущей странице окна документа> → [Ok]</p>

<i>1</i>	<i>2</i>
6. Обозначение объекта как активного	<Фигура “Роль классификатора”>: Db1Click → Класс → <установить IsActive> → [Ok]
7. Добавление имени и значений других свойств в фигуру «Несколько объектов»	<Фигура “Несколько объектов”>: Db1Click = → <Ввести имя! Значения свойств> → [Ok]
8. Управление отображением свойств в фигуре «Несколько объектов»	<Фигура “Несколько объектов”>: RClick → Параметры отображения фигуры → <установить!снять соответствующие флажки> → [Ok]
9. Создание пути роли «ассоциация»	Взаимодействия UML: 1 {<фигура ”Роль ассоциации”> → LClick_v → <расположить между фигурой ”Роль классификатора” и ”Несколько объектов”> → LClick_^ → <Левая конечная точка фигуры ”Роль ассоциации”> → LClick_v → <Точка соединения фигуры ”Роль классификатора”> → LClick_^ → <Правая конечная точка фигуры ”Роль ассоциации”> → LClick_v → <Точка соединения фигуры ”Несколько объектов”> → LClick_^ }*

1	2
10. Добавление сообщения к роли «ассоциация»	<Фигура "Роль ассоциации">: Db1Click = → <Ввести имя> → [Сообщения] → [Создать] = → <Ввести имя ! выражение для последовательности> → <Выбрать стереотип ! направление ! разновидность потока > → [Свойства...] → <Выбрать операцию [Создать (операцию)]> → 1 {[Ok]}*

3.7. Операции построения диаграмм классов

В Visio при помощи диаграммы классов не только строится статическая модель приложения, но и генерируется его исходный код, точнее, каркас приложения. Поэтому среди UML-диаграмм диаграмма классов является основной. Ниже в табл. 3.6 показаны основные операции, используемые при построении диаграммы классов, и последовательность выполнения каждой из них.

Операции построения диаграммы классов

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание структурной схемы	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели UML... → 1 { Статическая структура UML → 1 {<фигура "Класс"> → LClick_v → <Страница документа> → LClick_^ → <фигура "Объект"> → LClick_v → <Страница документа> → LClick_^ → <фигура {"Класс" "Объект"}> → DbClick => <атрибуты, операции> → [Ok] → RClick → Параметры отображения фигуры → <Установить требуемые флажки> → [Ok] → (п.3 6 11) → <фигура отношения {Ассоциация Ссылка Зависимость Обобщение Композиция}> → (аналогично п.3) → <фигура отношения> → DbClick => <оформление окончаний свойства> → [Ok]}*</p>

1	2
<p>2. Добавление интерфейса к классу, компоненту или другому элементу</p>	<p>Статическая структура UML: <фигура "Интерфейс"> → LClick_v → <Страница документа> → LClick_^ → <Конечная точка без кружка фигуры "Интерфейс"> → LClick_v → <Точка соединения компонента класса> → LClick_^ → <Фигура "Интерфейс"> → DbClick = → <Имя, операции, значения> → [Ok] (→ <Фигура "Интерфейс"> → RClick → (Показывать как интерфейс с описанием классов Показывать как интерфейс без описания операций) → [Ok] – интерфейс примет вид прямоугольной фигуры)</p>
<p>3. Отображение отношения реализации между классом и интерфейсом или другим элементом</p>	<p><Фигура {"Параметризованный класс" "Класс" "Служебная программа" "Метакласс"}>: RClick → Параметры отображения фигуры → Общие параметры → <установить флажок "Связь реализации"> → [Ok] → <Управляющий маркер связи "реализации" фигуры "Класс"> → LClick_v → <Точка соединения фигуры ("Интерфейс" "Класс" ...> → LClick_^</p>

1	2
<p>4. Добавление получений к классификатору</p>	<p><Фигура, выполняющая роль классификатора>: DbClick → {Получения [Создать]} → Получение ⇒ <Имя получения> → Сигнал ⇒ <из списка> ⇒ <Значения других свойств> → Исключения ⇒ <из списка> → {Ограничения Именованные значения} ⇒ 1 {Ok}*</p>
<p>5. Добавление шаблона к классу параметров</p>	<p><Фигура {”Параметризованный класс” ”Класс” ”Служебная программа” ”Метакласс” ”Связанный элемент”}>: DbClick → Параметры шаблона → ([Создать] Параметр шаблона ⇒ <Новое имя>) → Тип → [Нажать стрелку] → <Выбрать тип из списка> → Документация к... → <Добавить документацию> → [Ok]</p>

1	2
<p>6. Связывание параметров шаблона к классу</p>	<p>Статическая структура UML: 1 {<фигура “Связанный элемент”> → LClick_v → <Страница докумен- та (около фигуры ”Параметризо- ванный класс”)> → LClick_^ → <фигура “Привязка”> → LClick_v → <Страница документа> → LClick_v → <Конечная точка без наконечника фигуры ”Привязка”> → LClick_v → <Точка соединения фигуры ”Связанный элемент”> → LClick_^ → <Конечная точка с наконечником фигуры ”Привязка”> → LClick_v → <Точка соединения фигуры ”Параметризованный класс”> → LClick_^ → <фигура ”Привязка”> → DbClick → Свя- занные аргументы ⇒ (<Тип тре- буемому параметру> [Свойства] → <Тип значение > → [Ok]) → [Ok]}*</p>

1	2
<p>7. Создание объекта являющегося экземпляром класса</p>	<p>Статическая структура UML: 1 {<фигура “Класс”> → LClick_v → <Страница документа> → LClick_ ^ → (<Фигура “Класс”> → DbClick → Атрибуты → [Создать] → 1 {[Ok]}*) → <фигура Объект> → LClick_v → <Страница документа> → LClick_ ^ → DbClick → Объекты ⇒ <Ввести имя> → Класс → <Выбрать класс, экземпляром которого является данный объект> → 1 {Значения атрибута → <Требуемый атрибут> → [Свойства] ⇒ <Значение атрибута> → Связь атрибута ⇒ <Значения свойств> → [Ok]}* }*</p>
<p>8. Удаление параметров шаблона из класса</p>	<p><Фигура (“Параметризованный класс” “Класс” “Служебная программа” “Метакласс” “Связанный элемент”)>: DbClick → Параметры шаблона → 1 {<Выделить параметр> → [Удалить] → [Ok]}* → [Ok]</p>

1	2
<p>9. Изменение параметров шаблона класса</p>	<p><Фигура ("Параметризованный класс" "Класс" "Служебная программа" "Метакласс" "Связанный элемент")>: DblClick → Параметры шаблона → 1 {Параметр шаблона [Создать] => <Ввести имя> → Тип → [Нажать стрелку] → <Выбрать тип> → Документация к... => <Ввести документацию>}* → [Ok]</p>
<p>10. Скрытие раздела атрибутов и раздела операций класса</p>	<p><Фигура "Класс">: RClick → Параметры отображения фигуры → Скрыть → <установить флажок "Атрибуты" ("Операции")> → [Ok]</p>
<p>11. Отображение того, что на классы в пакете возможны ссылки извне</p>	<p>Статическая структура UML: 1 {<фигура "Зависимость"> → LClick_v → <Страница документа> → LClick_^ → <Конечная точка без наконечника фигуры "Зависимость"> → LClick_v → <Точка соединения ссылающегося пакета> → LClick_^ → <Конечная точка с наконечником фигуры "Зависимость"> → LClick_v → <Точка соединения пакета, на классы которого ссылка> → LClick_^ → <фигура "Зависимость"> → DblClick => <Ввести имя> → Стереотип → Импорт → [Ok]</p>

<i>1</i>	<i>2</i>
12. Отображение реализации типа при помощи реализующего класса	<p>Статическая структура UML:</p> <p>1 {<фигура “Класс”> → LClick_v → <Страница документа> → LClick_ ^ → DbClick → Класс ==> <Ввести имя> → Стереотип ==> <Выбрать тип Класс реализации> → [Ok]}* → <фигура Реализующего класса> → RClick → Параметры отображения фигуры → Общие параметры → <установить “Связь реализации”> → [Ok] → <фигура Реализующего класса> → <управляющий маркер фигуры> → LClick_v → <точка соединения фигуры “Класс”> → LClick_ ^</p>

3.8. Операции построения диаграммы компонентов

Особенности разработки диаграммы компонентов представлены в табл. 3.7, где перечислены операции, связанные с их построением, и последовательность выполнения каждой из этих операций.

Операции построения диаграммы компонентов

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание схемы компонентов	<p>Меню: Файл → Создать → Программное обеспечение... → Схема модели UML... → Схема компонентов → 1 {<фигура "Компонент"> → LClick_v → <Страница документа> → LClick_^ → <фигура "Интерфейс"> → LClick_v → <Страница документа> → LClick_^ → <Конечная точка без кружка фигуры "Интерфейс"> → LClick_v → <фигура "Компонент"> → LClick_^ → <фигура "Зависимость"> → LClick_v → <между зависимыми фигурами> → LClick_^ → (п.3) → <фигура ("Интерфейс" "Компонент")> → DbClick = → <имя ! атрибуты ! операции ! свойства> → [Ok]}*</p>
2. Добавление интерфейса к классу, компоненту или другому элементу	<p><фигура "Интерфейс">: <Конечная точка без кружка> → LClick_v → <Точка соединения компонента Клас-са > → LClick_^ → <фигура "Интерфейс"> → DbClick = → <имя ! операции ! свойства> → [Ok]</p>

<i>1</i>	<i>2</i>
3. Указание отношения зависимости между элементами	<фигура "Зависимость"> : <Конечная точка со стрелкой> → LClick_v → <Точка соединения главного элемента> → LClick_^ → <Конечная точка со стрелкой> → LClick_v → <Точка соединения зависимого элемента> → LClick_^ → <фигура "Зависимость"> → DbClick = → <имя ! стереотип ! свойства> → [Ok]

3.9. Операции построения диаграммы развертывания

Диаграммы развертывания (размещения) приведены в табл. 3.8, где даны операции, связанные с их построением, и последовательность выполнения каждой из этих операций.

Таблица 3.8

Операции построения диаграммы развертывания

Операция	Рекомендуемая последовательность действий. Примечания
<i>1</i>	<i>2</i>
1. Создание топологической схемы	Меню: Файл → Создать → Программное обеспечение... → Схема модели UML... → Топологическая схема → 1 {<фигура "Узел" ("Компонент" "Объект")> → LClick_v → <Страница документа> → LClick_^ → 1 {(п.2)}* → <фигура ("Интерфейс" "Сообщение" "Зависимость")> → (п.3 4 5)}*

1	2
2. Включение элементов в узел на схеме	<p><фигура ("Узел" "Экземпляр узла")>: DbfClick = → <имя ! атрибуты ! операции ! свойства> → [Ok] → <Угловой маркер> → Lfclick_v → <включение элементов в контур> → Lfclick_^ → <фигура ("Компонент" ! "Экземпляр компонента" ! "Объект")> → Lfclick_v → <поверх фигуры "Узел"> → Lfclick_^ → 1 {<фигура "Зависимость"> → Lfclick_v → <зависимые фигуры> → Lfclick_^ }* → 1 {<фигура ("Узел" "Компонент"> → DbfClick → ([Компоненты] [Узлы]) → <соответствующие {компоненты данного узла узлы данного компонента}> → [Ok] }* }</p>
3. Добавление интерфейса к классу, компоненту или другому элементу	<p><фигура "Интерфейс">: Lfclick_v → <Страница документа> → Lfclick_^ → <конечная точка без кружка фигуры "Интерфейс"> → Lfclick_v → <точка соединения фигуры компонента> → Lfclick_^ → DbfClick = → <имя ! операции ! свойства> → [Ok]</p>
4. Задать ассоциацию сообщения между узлами	<p><фигура "Сообщение">: <Конечная точка1> → Lfclick_v → <точка соединения первого узла> → Lfclick_^ → <Конечная точка2 > → Lfclick_v → <точка соединения второго узла> → Lfclick_^ → <фигура "Сообщение"> → DbfClick = → <имя ! свойства> → [Ok]</p>

1	2
5. Задание отношения зависимости между элементами	<p><фигура "Зависимость">: <Конечная точка со стрелкой> → LClick_v → <Точка соединения главного элемента> → LClick_^ → <Конечная точка со стрелкой> → LClick_v → <Точка соединения зависимого элемента> → LClick_^ → <фигура "Зависимость"> → DbClick = → <имя ! стереотип ! свойства> → [Ok]</p>

Следует отметить, что в части построения диаграмм развертывания графический редактор Visio превосходит CASE – средство Rational Rose. Visio позволяет совместить диаграмму развертывания с диаграммой компонентов. К тому же значки узлов можно показать более оригинально, что допускается синтаксисом UML.

СПИСОК ЛИТЕРАТУРЫ

1. Бугай, О.В. Системы автоматического проектирования программного обеспечения: учебное пособие для студентов специальности 1-40 01 02-03 «Информационные системы и технологии» / О.В. Бугай, В.С. Юденков. – Минск: БГТУ, 2007. – 172 с.
2. Буч, Г. Язык UML: Руководство пользователя / Г. Буч, Д. Рамбо, А. Джекобсон; пер. с англ. – М.: ДМК, 2000. – 432 с.
3. Калянов, Г.Н. CASE: структурный системный анализ (автоматизация и применение) / Г.Н. Калянов. – М.: ЛОРН, 1996. – 242 с.
4. Калянов, Г.Н. CASE-технологии. Консалтинг в автоматизации бизнес-процессов / Г.Н. Калянов. – 3-е изд. – М.: Горячая линия – Телеком, 2002. – 320 с.
5. Карпов, Б. Microsoft VISIO 2000: краткий курс / Б. Карпов, Н. Мирошниченко. – СПб.: Питер, 2001. – 256 с.
6. Леоненков, А.В. Самоучитель UML / А.В. Леоненков. – СПб.: БХВ-Петербург, 2001. – 304 с.
7. Фаулер, М. UML. Основы / М. Фаулер, К. Скотт; пер. с англ. – СПб.: Символ-Плюс, 2002. – 192 с.
8. Харрингтон, Д. Проектирование реляционных баз данных. Просто и доступно / Д. Харрингтон. – М.: Лори, 2002. – 230 с.
9. Шмуллер, Дж. Освой самостоятельно UML за 24 часа / Дж. Шмуллер; пер. с англ. – 2-е изд. – М.: Вильямс, 2002. – 352 с.