

ЛАБОРАТОРНАЯ РАБОТА

Разработка и построение диаграмм классов

Цель работы: ознакомиться с требованиями, предъявляемыми к построению диаграмм классов; научиться строить диаграммы классов, используя специализированные программные комплексы.

Теоретическая часть

Вершины в диаграммах классов

Итак, вершина в диаграмме классов — класс. Обозначение класса показано на рис. 11.1.

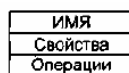


Рис. 11.1. Обозначение класса

Имя класса указывается всегда, свойства и операции — выборочно. Предусмотрено задание области действия свойства (операции). Если свойство (операция) подчеркивается, его областью действия является класс, в противном случае областью Действия является экземпляр (рис. 11.2).

Что это значит? Если областью действия свойства является класс, то все его экземпляры (объекты)

используют общее значение этого свойства, в противном случае у каждого экземпляра свое значение свойства.

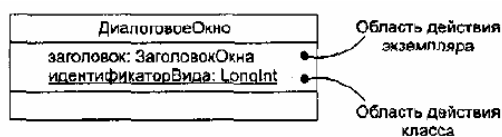


Рис. 11.2. Свойства уровней класса и экземпляра

Свойства

Общий синтаксис представления свойства имеет вид

Видимость Имя [Множественность]: Тип = НачальноеЗначение {Характеристики}

Рассмотрим видимость и характеристики свойств.

В языке UML определены три уровня видимости:

public	Любой клиент класса может использовать свойство (операцию), обозначается символом +
protected	Любой наследник класса может использовать свойство (операцию), обозначается символом #
private	Свойство (операция) может использоваться только самим классом, обозначается символом -

ПРИМЕЧАНИЕ

Если видимость не указана, считают, что свойство объявлено с публичной видимостью.

Определены три характеристики свойств:

changeable	Нет ограничений на модификацию значения свойства
addOnly	Для свойств с множественностью, большей единицы; дополнительные значения могут быть добавлены, но после создания значение не может удаляться или изменяться
frozen	После инициализации объекта значение свойства не изменяется

ПРИМЕЧАНИЕ

Если характеристика не указана, считают, что свойство объявлено с характеристикой changeable.

Примеры объявления свойств:

начало	Только имя
+ начало	Видимость и имя
начало : Координаты	Имя и тип
имяфамилия [0..1] : String	Имя, множественность, тип
левыйУгол : Координаты=(0, 10)	Имя, тип, начальное значение
сумма : Integer {frozen}	Имя и характеристика

Операции

Общий синтаксис представления операции имеет вид

Видимость Имя (Список Параметров): ВозвращаемыйТип {Характеристики}

Примеры объявления операций:

записать	Только имя
+ записать	Видимость и имя
зарегистрировать) и: Имя, ф: Фамилия)	Имя и параметры
балансСчета () : Integer	Имя и возвращаемый тип
нагревать () (guarded)	Имя и характеристика

В сигнатуре операции можно указать ноль или более параметров, форма представления параметра имеет следующий синтаксис:

Направление Имя : Тип = ЗначениеПоУмолчанию
 Элемент Направление может принимать одно из следующих значений:

in	Входной параметр, не может модифицироваться
out	Выходной параметр, может модифицироваться для передачи информации в вызывающий объект
inout	Входной параметр, может модифицироваться

Допустимо применение следующих характеристик операций:

leaf	Конечная операция, операция не может быть полиморфной и не может переопределяться (в цепочке наследования)
isQuery	Выполнение операции не изменяет состояния объекта
sequential	В каждый момент времени в объект поступает только один вызов операций. Как следствие, в каждый момент времени выполняется только одна операция объекта.
guarded	Другими словами, допустим только один поток вызовов (поток управления) Допускается одновременное поступление в объект нескольких вызовов, но в каждый момент времени обрабатывается только один вызов охраняемой операции. Иначе говоря, параллельные потоки управления исполняются последовательно (за счет постановки вызовов в очередь)
concurrent	В объект поступает несколько потоков вызовов операций (из параллельных потоков управления). Разрешается параллельное (и множественное) выполнение операции. Предполагается, что такие операции являются атомарными

Организация свойств и операций

Известно, что пиктограмма класса включает три секции (для имени, для свойств и для операций). Пустота секции не означает, что у класса отсутствуют свойства или операции, просто в данный момент они не показываются. Можно явно определить наличие у класса большого количества свойств или атрибутов. Для этого в конце показанного списка проставляются три точки. Как показано на рис. 11.3, в длинных списках свойств и операций разрешается группировка — каждая группа начинается со своего стереотипа.

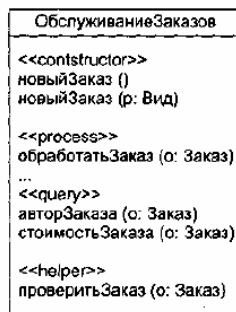


Рис. 11.3. Стереотипы для характеристик класса

Множественность

Иногда бывает необходимо ограничить количество экземпляров класса:

- задать ноль экземпляров (в этом случае класс превращается в утилиту, которая предлагает свои свойства и операции);
- задать один экземпляр (класс-singleton);
- задать конкретное количество экземпляров;
- не ограничивать количество экземпляров (это случай, предполагаемый по умолчанию).

Количество экземпляров класса называется его множественностью. Выражение множественности записывается в правом верхнем углу значка класса. Например, как показано на рис. 11.4,

КонтроллерУглов — это класс-singleton, а для класса ДатчикУгла разрешены три экземпляра.

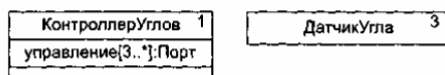


Рис. 11.4. Множественность

Множественность применима не только к классам, но и к свойствам. Множественность свойства задается выражением в квадратных скобках, записанным после его имени. Например, на рисунке заданы три и более экземпляра свойства Управление (в экземпляре класса КонтроллерУглов).

Практическая часть

Разработать и построить диаграмму классов в соответствии с вариантом.

Порядок выполнения лабораторной работы:

1. Получить вариант разрабатываемой диаграммы у преподавателя.
2. Разработать и построить диаграмму классов в соответствии с требованиями оформления, используя специализированную программу.
3. Защитить лабораторную работу:
 - 3.1 Сдать правильно оформленную диаграмму.
 - 3.2 Ответить на контрольные вопросы.

Контрольные вопросы

1. Для чего применяются диаграммы классов?
2. Какие предметы используются при построении диаграмм классов?
3. Какие отношения используются при построении диаграмм классов?
4. Назовите основные этапы построения диаграмм классов.
5. Ограничения применимости диаграмм классов.

Литература

1. Технологии разработки программного обеспечения: Учебник/ С. Орлов. — СПб.: Питер, 2002. — 464 с.: ил.