

# ПИТОН

## Курс лекций

### Лекция четырнадцатая

© Зайцев Вадим Валерьевич, 2002–2010,  
[spider.vz@gmail.com](mailto:spider.vz@gmail.com)

#### 6.5 Иерархия

Последняя главная составляющая объектного подхода — это, конечно же, иерархия. Предыдущая тема могла немного сбить нас с основного курса, поэтому сделаем предварительные выводы. Ни один программист, даже самый гениальный, не в состоянии разобраться в большом проекте, если будет его рассматривать целиком во всех подробностях. Для декомпозиции большого проекта используется абстракция, которая выделяет определённую часть системы, унифицируя её с реальным объектом. Помощь инкапсуляции также может быть легко облечена в слова: инкапсуляция прячет незначительные, запутывающие, отвлекающие детали абстракции. Таким образом, мы сначала смотрим на систему в целом, затем определяем границы её составных частей, затем даём каждой части имя и подменяем рассмотрение части лицезрением имени (и его внешних свойств, разумеется). Модульность позволяет зайти с другого края, выделив в системе подсистемы.

Существуют системы настолько крупные, что число абстракций в них превышает пределы человеческих перцептивных возможностей. При этом они могут быть тесно связаны, лишая программиста возможности применения модульной декомпозиции. Рассмотренные нами подходы не дадут ответа на вопрос: «Что же делать в таком случае?» Ответ — в следующем определении.

*Иерархия — это упорядочение абстракций, расположение их по уровням.*

Этот метод, позволяющий рассматривать разнородные абстракции по отдельности, есть великое изобретение голландского математика Эдсгера Дейкстры и просто находка для объектного подхода. Иерархия не только позволяет рассматривать, исследовать различные совокупности абстракций, но и разделить собственно программирование на логически завершённые этапы, на каждом из которых реализуются абстракции определённого уровня.

Поясним идею подразделения абстракций на подчинённые друг другу группы на примере технологии написания современных крупных 3D-проектов: мультфильмов, спецэффектов и т.п. Будем считать сценарий уже написанным и все технические проблемы решёнными.

Сначала группа художников просто рисует, как в том или ином ракурсе выглядит описываемый предмет. Затем группа модельщиков на основе этих входных данных создаёт статичные трёхмерные объекты, используя примитивные методы, уже содержащиеся в программе для моделирования (мэппинг, рендеринг, текстурирование, шейдинг, фильтрация, антиалиасинг, эффекты коррекции света и др.). Затем описывают более сложные функции: перемещение камеры вокруг предмета, изменение освещённости, простейшие движения самого предмета, его частей. Следующим этапом стоит написание крупных процедур, реализующих движение предмета: как он делает шаг, как хватает другой предмет, как говорит, как падает. . . Одновременно начинают работу озвучиватели, стремящиеся добиться максимальной отдачи и реального трёхмерного звука с помощью реверберации, акклюзий, обструкций, морфинга, позиционирования и прочих готовых продуктов. Следующий этап ещё сложнее, чаще всего он выполняется с привлечением не только художников, но и актёров. Он включает описание таких процессов, как ходьба, произнесение длинной речи с жестикულიацией, взаимодействие предметов друг с другом и с окружающей средой и т.п. Финальным этапом является готовый фильм (или сцена).

Таким образом, мы своим непрофессиональным взглядом насчитали 6 этапов, каждый следующий из которых построен из предыдущего. Процедуры, реализующиеся на различных этапах, принадлежат разным уровням абстракции. Мы не обязаны думать о поведении каждого пикселя, делая шаг и о каждом шаге, моделируя бег большой толпы.

Вернувшись к теории объектно-ориентированного подхода (кстати, повысив при этом уровень абстракции), мы можем дать ответ на мучивший нас не так давно вопрос. Что делать, если система слишком сложна для понимания, несмотря на то, что все подходы были применены верно? Полный ответ таков: добавить уровень абстракции и рассматривать каждый уровень отдельно от других. Сложные действия куда проще описывать через более мелкие действия, чем писать их заново на каждом этапе.

Совершенство иерархии заключается в том, что уровни абстракции можно добавлять потенциально бесконечно. На практике обычно довольствуются их конечным числом.

Основными типами иерархических структур в сложных системах являются иерархия «является», иерархия «используется» и иерархия «часть».