

**К вопросу выбора презентативности  
модели ПО при его разработке**

Бугай О.В., Лапанович И.О.

Белорусский национальный технический университет

С усложнением ПО многие его разработчики были вынуждены перейти от известного метода «собачьей конуры» к методологиям, предусматривающим предварительное проектирование программ с последующим созданием их исходного кода. В отличие от проектов, воплощенных у строителей, машиностроителей и других в виде чертежей, проект ПО обычно представляют его моделями, реализуемых рядом диаграммных техник и поддержанных такими инструментальными средствами как VISIO, BPwin, ERwin, Rational Rose и т.п.

Если в строительстве с тысячелетней историей, машиностроении, насчитывающем несколько столетий, вопрос о требованиях к проекту можно считать закрытым, то в случае с программированием отношение к полноте модели ПО далеко не однозначно. Одни группы разработки ПО предваряют этапы написания исходного кода системы либо подсистемы исчерпывающим ее моделированием, другие – вообще игнорируют его. Следствием этого явилось появление множества ТР ПО, в частности, объектно-ориентированной – ОР, UP, RUP, XP и других.

На наш взгляд на сложившуюся ситуацию влияет целый ряд обстоятельств. Вот основные из них:

- Особенности создания ПО – один экземпляр создается, а остальные тиражируются. Если учесть тот факт, что в этом случае в отличие от изготовления промышленных изделий, модель ПО больше не нужна, то его влияние на отношение к моделированию может быть весьма существенным.
- Особенности разделения труда между разработчиками – очень редко ПО проектирует одна команда, а реализует другая. При наличии двух команд модель ПО должна быть более обстоятельной, чем в

ситуации, когда проектирование ПО и его реализация выполняются одной и той же командой.

- Сложность предмета разработки – простые программы вообще нет смысла моделировать. Постоянное усложнение ПО и заставило его разработчиков прибегнуть к созданию проекта программ в виде модели.
- Стиль программирования – при структурном системном подходе в случае разработки сложного ПО обстоятельная модель необходима, а при объектно-ориентированном – все зависит от возможности повторного использования кодов, ранее разработанных компонент.
- Направленность тематики – создание программ по освоенной командной тематике не требует детальной проработки модели.
- Перспектива модификации – очевидно, что ПО, которое предполагается в будущем модифицировать должно более скрупулезно проектироваться.
- Позиция заказчика – если заказчик нуждается в исчерпывающе-сопроводительной документации, разработчик вынужден включить модель ПО вне зависимости насколько она была ему необходима в процессе выполнения заказа.
- ИСР – оснащенность интегрированной среды разработки компонентами весьма существенно влияет на требования к проекту разрабатываемого ПО. При наличии в интегрированной среде разработки необходимых команде разработчиков компонент можно ограничиться высокоуровневым представлением модели. Следует при этом иметь в виду, что любая ИСР навязывает программистам свой стиль решения задачи.

Кроме перечисленных выше факторов на отношение к моделированию несомненно влияют инженерные наклонности разработчиков: программисты с инженерным мышлением предпочтут начать разработку ПО с построения модели, а математики будут всячески избегать его (математическое моделирование здесь не в счет). С учетом того, что основной

контингент программистов составляют математики, то вполне понятно стремление к снижению доли проектирования при разработке ПО. Тем более, что широкие возможности современных ИСР способствуют этому. В результате наблюдаем повышенный интерес к экстремальному программированию.

Да и существующие СУБД приучают разработчика стартовать не от модели БД, а – наоборот. Например, в строительстве ситуация, когда по мере возведения здания создавались бы чертежи, демонстрирующие достигнутое, считалась бы абсурдной, а вот при работе с СУБД – воспринимается вполне нормально и комизм ситуации остается не замеченным.

Как видим, полярное отношение к моделированию ПО при его разработке имеет вполне определенное обоснование и каждый из подходов к вопросу проектирования рассматриваемого предмета разработки имеет право на существование. Тем более, что работа над многими программами по своей специфике сродни изобретательской деятельности.

### **Литература**

1. Буг, Г., Рамбо, Дж., Джекобсон, А. Язык UML. Руководство пользователя: Пер. с англ. – М.: ДМК, 2000. – 432 с.
2. Мартин, Робертс, С. Быстрая разработка программ: принципы, примеры, практика: Пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 752 с.
3. Фаулер, М. Разработка ПО: проектирования больше нет?! Пер. с англ. К. Максимова, А. Максимовой// Сетевые решения. – 2004. - №1. – с.78-86.