

Евгений Шуремов

---

*Методологические подходы  
и средства поддержки  
процессов разработки  
программного обеспечения  
организационно-  
экономических систем*

---

Коротко о главном

---

**Евгений Леонидович Шуремов**  
**Методологические подходы**  
**и средства поддержки**  
**процессов разработки**  
**программного обеспечения**  
**организационно-экономических**  
**систем. Коротко о главном**

*[http://www.litres.ru/pages/biblio\\_book/?art=22572757](http://www.litres.ru/pages/biblio_book/?art=22572757)*

*ISBN 9785448360961*

**Аннотация**

Книга содержит краткое изложение основной проблематики, связанной с процессами промышленной разработки программного обеспечения информационных систем организационно-экономического управления. Рассматриваются основные существующие методологические подходы и методики организации разработки, а также наиболее популярные поддерживающие их инструментальные средства. Изложение ориентировано на минимально подготовленных читателей, желающих получить общее представление по данной теме.

# Содержание

Введение	5
Роль программного обеспечения в развитии организационно-экономических систем	7
Программное обеспечение как объект развития	11
Конец ознакомительного фрагмента.	23

**Методологические подходы  
и средства поддержки  
процессов разработки  
программного обеспечения  
организационно-  
экономических систем**

**Коротко о главном**

**Евгений Шуремов**

© Евгений Шуремов, 2017

ISBN 978-5-4483-6096-1

Создано в интеллектуальной издательской системе Ridero

# Введение

Данная публикация содержит краткое изложение основной проблематики, связанной с процессами промышленной разработки программного обеспечения информационных систем организационно-экономического управления. Рассматриваются основные существующие методологические подходы и методики организации разработки, а также наиболее популярные поддерживающие их инструментальные средства.

Изложение ориентировано на минимально подготовленных читателей, желающих получить общее представление по данной теме. В основу книги положены тезисы лекций, на протяжении нескольких лет читавшихся автором для магистрантов Финансового университета при Правительстве РФ.

Книга имеет Интернет-поддержку на сайте <http://shurem.ru>. На страницах поддержки будут публиковаться уточнения и дополнительные материалы к тексту книги. Кроме того, там будут размещаться средства автоматизированного самотестирования на знание основных положений представленного в книге материала, а также инструкции по его проведению. Однако страницы Интернет-поддержки доступны только авторизованным пользователям сайта <http://shurem.ru>. «Случайным» посетителям сайта эти стра-

ницы недоступны. То есть необходимо зарегистрироваться, авторизоваться на сайте и далее пройти по пути **Публикации => Поддержка книг => Методологические подходы и средства поддержки процессов разработки программного обеспечения организационно-экономических систем.**

С пожеланиями и предложениями можно обратиться к автору по адресу [shurem@mail.ru](mailto:shurem@mail.ru)

# **Роль программного обеспечения в развитии организационно- экономических систем**

В организационно-экономических информационных системах (ИС) процедуры информационного процесса интегрированы с функциями управления. Наряду со своими основными функциями, их непосредственно выполняет управленческий персонал. Поэтому в современных условиях ИС следует рассматривать как неотъемлемую часть инфраструктуры организационно-экономической системы, инструмент решения всего комплекса задач управления организацией.

Развитие любой организации так или иначе требует развития ее управленческой инфраструктуры. Отсюда следует, что ИС постоянно нуждаются в развитии. В условиях информационной экономики это направление развития считается приоритетным по отношению к другим направлениям.

Далее (если отдельно не оговорено противное) рассматриваются ИС организационно-экономического управления.

Бурное развитие информационных технологий постоянно расширяет вариативность выбора стратегий развития ИС.

**50—70 гг.** ИС (АСУ) на базе мэйнфреймов. Создавались в крупных организациях для решения отдельных задач управления, требующих массового ввода и оперативной об-

работки больших объемов информации. Полностью централизованная обработка данных в вычислительных центрах. Выбор вариантов развития ИС ограничен возможностями приобретения новых вычислительных мощностей на мэйнфреймах и разработкой для них программ для автоматизации решения новых задач.

**70—80 гг.** ИС на базе мини-ЭВМ. Создавались на средних и даже относительно небольших предприятиях для решения широкого круга задач. Выбор шире: можно создать ИС на основе мэйнфреймов или на базе мини-ЭВМ. При этом выбор технической платформы предопределяет возможность использования программного обеспечения (либо для одной, либо для другой аппаратной платформы).

**80 гг.** ИС с распределенной обработкой данных на персональных компьютерах. Массовое распространение офисных программ. Начало формирования электронного документооборота. Выбор еще шире: можно по-старинке создать ИС на базе мэйнфреймов или мини-ЭВМ; полностью распределенную систему обработки данных на ПК; часть задач решать на мини-ЭВМ, а часть на ПК; для ПК можно выбрать программы разных поставщиков и т. д.

**80—90 гг.** Создание и массовое распространение ERP-систем. Массовый переход к клиент-серверным технологиям. Выбор еще шире: наряду с возможностями предыдущих этапов, можно выбрать разные варианты организации локальной сети, включая разделение доступа к периферийным



устройствам; разные технологии функционирования прикладных программ в сети; разное обеспечивающее и прикладное ПО и т. д.

**Середина 90 – конец нулевых гг.** Широкое распространение электронного документооборота. Развитие электронной коммерции и коммуникаций с сетями внешних партнёров и клиентов через Интернет (B2B, B2C, SCM, CSRP и др.). Дополнительно к предыдущему: разные способы взаимодействия локальных сетей разных организаций через Интернет; выбор стандартов и форматов обмена данными; дальнейшее расширение возможностей прикладных программ; выбор способов взаимодействия сотрудников через глобальные сети и т. д.

**Настоящее время.** Широкое распространение мобильных устройств. Рост популярности «облачных» вычислений. Дополнительно к предыдущему: выбор вариантов взаимодействия с мобильными пользователями (сотрудниками, партнерами, клиентами); иметь собственные вычислительные мощности или арендовать на стороне; использовать «монолитную» систему обработки данных или набор web-сервисов и т. д.

Различные стратегии развития ИС могут требовать разных затрат, иметь следствием разную *совокупную стоимость* владения системой, обеспечивать разный уровень отдачи на инвестиции. Однако какой бы вариант развития ИС ни был выбран, ключевым элементом развития является уве-

лишение функциональности ИС, возможность автоматизации решения более широкого круга задач, чем раньше. А поскольку функциональность ИС обеспечивается, прежде всего, ее программным обеспечением, то при любых условиях развитие программного обеспечения является важнейшим элементом развития любой ИС.

# Программное обеспечение как объект развития

**Программное обеспечение** (ПО) – все или часть программ, процедур, правил и соответствующей документации системы обработки информации (ISO/IEC 2382—1: 1993. Information technology – Vocabulary – Part 1: Fundamental terms).

Другие определения из международных и отечественных стандартов:

Компьютерные программы, процедуры и, возможно, соответствующая документация и данные, относящиеся к функционированию компьютерной системы (FCD ISO/IEC 24765. Systems and Software Engineering Vocabulary).

Совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ (ГОСТ 19781—90).

Программное обеспечение является объектом изучения информатики, программирования и программной инженерии.

Термин **software** впервые применил математик из Принстонского университета Джон Тьюки (англ. John W. Tukey) в статье в American Mathematical Monthly в 1958 году.

Первая теория, касающаяся программного обеспечения, была предложена английским математиком Аланом Тью-

рингом в 1935 году в эссе «Computable numbers with an application to the Entscheidungsproblem (Decision problem)». Он создал так называемую машину Тьюринга, математическую модель абстрактной машины, способной выполнять последовательности рудиментарных операций, которые переводят машину из одного фиксированного состояния в другое. Главная идея заключалась в математическом доказательстве факта, что любое наперёд заданное состояние системы может быть всегда достигнуто последовательным выполнением конечного набора элементарных команд (программы) из фиксированного набора команд.

**Архитектура программного обеспечения** – структура программы или вычислительной системы, которая включает программные компоненты, видимые снаружи свойства этих компонентов, а также отношения между ними.

**Качество программного обеспечения** – весь объем признаков и характеристик программ, который относится к их способности удовлетворять установленным или предполагаемым потребностям.

**Тестирование программного обеспечения** – процесс исследования программного обеспечения (ПО) с целью получения информации о качестве продукта.

С точки зрения **ISO 9126**, качество программного обеспечения можно определить как совокупную характеристику исследуемого ПО с учётом следующих составляющих:

– Надёжность;

- Сопровождаемость;
- Практичность;
- Эффективность;
- Мобильность;
- Функциональность.

**Лицензия на программное обеспечение** – правовой инструмент, определяющий использование и распространение программного обеспечения, защищённого авторским правом.

Обычно лицензия на программное обеспечение разрешает получателю использовать одну или несколько копий программы, причём без лицензии такое использование рассматривалось бы в рамках закона как нарушение авторских прав издателя. По сути, лицензия выступает гарантией того, что издатель ПО, которому принадлежат исключительные права на программу, не подаст в суд на того, кто ею пользуется.

Математические и алгоритмические методы, содержащиеся в программном обеспечении, могут быть запатентованы.

По определению, предложенному Фондом за свободную информационную инфраструктуру, программный патент – это «патент на что-либо, выполняемое компьютером посредством программного обеспечения».

Защитники программных патентов считают, что они позволяют:

- защитить сложное ПО от подражателей, которым не нужно тратить время и деньги на проектные работы;

– защитить изобретателей-одиночек от крупных компаний;

– труднодоступность запатентованных технологий стимулирует создание более совершенных технологий.

**Документация** – печатные руководства пользователя, диалоговая (оперативная) документация и справочный текст, описывающие как пользоваться программой.

Документация состоит из отдельных документов.

**Документ как элемент документации** – это целевая информация, предназначенная для конкретной аудитории, размещённая на конкретном носителе в заданном формате.

**Программный документ** – документ, содержащий в зависимости от назначения данные, необходимые для разработки, производства, эксплуатации и сопровождения программы.

**Разработка программного обеспечения** (software development) – это процесс, направленный на создание и поддержание работоспособности, качества и надёжности программного обеспечения.

Как и традиционные инженерные дисциплины, разработка программного обеспечения имеет дело с проблемами качества, стоимости и надёжности. Некоторые программы содержат миллионы строк исходного кода, которые должны правильно исполняться в изменяющихся условиях. Сложность ПО сравнима со сложностью наиболее совершенных из современных машин, таких как самолёты.

Накопленный к настоящему времени опыт создания программных систем (ПС) показывает, что это сложная и трудоемкая работа, требующая высокой квалификации участвующих в ней специалистов. Однако до настоящего времени создание таких систем нередко выполняется на интуитивном уровне с применением неформализованных методов, основанных на искусстве, практическом опыте, экспертных оценках и дорогостоящих экспериментальных проверках качества функционирования ПО. По данным Института программной инженерии (Software Engineering Institute, SEI) в последние годы до 80% всего эксплуатируемого ПО разрабатывалось вообще без использования какой-либо дисциплины проектирования, методом «code and fix» (кодирования и исправления ошибок).

Проблемы создания ПО следуют из его свойств. Еще в 1975 г. Фредерик Брукс, проанализировав свой уникальный по тем временам опыт руководства крупнейшим проектом разработки операционной системы OS/360, определил перечень неотъемлемых свойств ПО: сложность, согласованность, изменяемость и незримость.

Современные крупномасштабные проекты создания и развития ПС характеризуются следующими особенностями.

### **Характеристики объекта внедрения:**

– структурная сложность (многоуровневая иерархическая структура организации) и территориальная распреде-

ленность;

- функциональная сложность (многоуровневая иерархия и большое количество функций, выполняемых организациями; сложные взаимосвязи между ними);

- информационная сложность (большое количество источников и потребителей информации (министерства и ведомства, местные органы власти, организации-партнеры), разнообразные формы и форматы представления информации, сложная информационная модель объекта – большое количество информационных сущностей и сложные взаимосвязи между ними), сложная технология прохождения документов;

- сложная динамика поведения, обусловленная высокой изменчивостью внешней среды (изменения в законодательных и нормативных актах, нестабильность экономики и политики) и внутренней среды (структурные реорганизации, текучесть кадров).

### **Технические характеристики проектов создания ПО:**

- различная степень унифицированности проектных решений в рамках одного проекта;

- высокая техническая сложность, определяемая наличием совокупности тесно взаимодействующих компонентов (подсистем), имеющих свои локальные задачи и цели функционирования (транзакционных приложений, предъявляющих повышенные требования к надежности, безопасности



и производительности, и приложений аналитической обработки (систем поддержки принятия решений), использующих нерегламентированные запросы к данным большого объема);

- отсутствие полных аналогов, ограничивающее возможность использования каких-либо типовых проектных решений и прикладных систем, высокая доля вновь разрабатываемого ПО;

- большое количество и высокая стоимость унаследованных приложений (существующего прикладного ПО), функционирующих в различной среде (персональные компьютеры, миникомпьютеры, мэйнфреймы), необходимость интеграции унаследованных и вновь разрабатываемых приложений;

- большое количество локальных объектов внедрения, территориально распределенная и неоднородная среда функционирования (СУБД, операционные системы, аппаратные платформы);

- большое количество внешних взаимодействующих систем – различных организаций с различными форматами обмена информацией (налоговая служба, налоговая полиция, Госстандарт, Госкомстат, Министерство финансов, МВД, местная администрация).

**Организационные характеристики проектов создания ПО:**

- различные формы организации и управления проек-

том: централизованно управляемая разработка тиражируемого ПО, экспериментальные пилотные проекты, инициативные разработки, проекты с участием как собственных разработчиков, так и сторонних компаний на контрактной основе;

- большое количество участников проекта как со стороны заказчиков (с разнородными требованиями), так и со стороны разработчиков (более 100 человек), разобщенность и разнородность отдельных групп разработчиков по уровню квалификации, сложившимся традициям и опыту использования тех или иных инструментальных средств;

- значительная длительность жизненного цикла системы, в том числе значительная временная протяженность проекта, обусловленная масштабами организации-заказчика, различной степенью готовности отдельных ее подразделений к внедрению ПО и нестабильностью финансирования проекта;

- высокие требования со стороны заказчика к уровню технологической зрелости организаций-разработчиков (наличие сертификации в соответствии с международными и отечественными стандартами).

Уже в конце 60-х годов прошлого века в США было отмечено явление под названием «**software crisis**» (кризис ПО). Это выражалось в том, что большие проекты стали выполняться с отставанием от графика или с превышением сметы расходов, разработанный продукт не обладал требуемы-

ми функциональными возможностями, производительность его была низка, качество получаемого программного обеспечения не устраивало потребителей.

Аналитические исследования и обзоры, выполняемые в течение ряда последних лет ведущими зарубежными аналитиками, показывали не слишком обнадеживающие результаты. Так, например, результаты исследований, выполненных в 1995 году компанией Standish Group, которая проанализировала работу 364 американских корпораций и итоги выполнения более 23 тысяч проектов, связанных с разработкой ПО, выглядели следующим образом:

- только 16,2% завершились в срок, не превысили запланированный бюджет и реализовали все требуемые функции и возможности;
- 52,7% проектов завершились с опозданием, расходы превысили запланированный бюджет, требуемые функции не были реализованы в полном объеме;
- 31,1% проектов были аннулированы до завершения;
- для двух последних категорий проектов бюджет среднего проекта оказался превышенным на 89%, а срок выполнения – на 122%.

В соответствии с исследованиями 1998 года процентное соотношение трех перечисленных категорий проектов лишь немного изменилось в лучшую сторону (26%, 46% и 28% соответственно).

В последние годы процентное соотношение трех перечис-

ленных категорий проектов также незначительно изменяется в лучшую сторону, однако, по оценкам ведущих аналитиков, это происходит в основном за счет снижения масштаба выполняемых проектов, а не за счет повышения управляемости и качества проектирования.

В числе причин возможных неудач, по мнению разработчиков, фигурируют:

- нечеткая и неполная формулировка требований к ПО;
- недостаточное вовлечение пользователей в работу над проектом;
- отсутствие необходимых ресурсов;
- неудовлетворительное управление проектом;
- частое изменение требований и спецификаций;
- новизна и несовершенство используемой технологии;
- недостаточная поддержка со стороны высшего руководства;
- недостаточно высокая квалификация разработчиков, отсутствие необходимого опыта.

Объективная потребность контролировать процесс разработки сложных систем ПО, прогнозировать и гарантировать стоимость разработки, сроки и качество результатов привела в конце 60-х годов прошлого века к необходимости перехода от кустарных к индустриальным способам создания ПО и появлению совокупности инженерных методов и средств создания ПО, объединенных общим названием «программная инженерия» (**software engineering**). В основе программной

инженерии лежит следующая фундаментальная идея: проектирование ПО является формальным процессом, который можно изучать и совершенствовать. Освоение и правильное применение методов и средств создания ПО позволяет повысить его качество, обеспечить управляемость процесса проектирования ПО и увеличить срок его жизни.

В то же время, попытки чрезмерной формализации процесса, а также прямого заимствования идей и методов из других областей инженерной деятельности (строительства, производства) привели к ряду серьезных проблем. После двух десятилетий напрасных ожиданий повышения продуктивности процессов создания ПО, возлагаемых на новые методы и технологии, специалисты в индустрии ПО пришли к пониманию, что фундаментальная проблема в этой области – **неспособность эффективного управления проектами создания ПО**.

Выяснилось, что невозможно достичь удовлетворительных результатов от применения даже самых совершенных технологий и инструментальных средств, если они применяются бессистемно. Часто разработчики не обладают необходимой квалификацией для работы с ними, а сам проект выполняется и управляется хаотически, в режиме «тушения пожара». Бессистемное применение технологий создания ПО (ТС ПО), в свою очередь, порождает разочарование в используемых методах и средствах. Анализ мнений разработчиков показывает, что среди факторов, влияющих на эф-

эффективность создания ПО, используемым методам и средствам придается гораздо меньшее значение, чем квалификации и опыту разработчиков. Если в таких условиях отдельные проекты завершаются успешно, то этот успех достигается за счет героических усилий фанатично настроенного коллектива разработчиков. Постоянное повышение качества создаваемого ПО и снижение его стоимости может быть обеспечено только при условии достижения организацией необходимой технологической зрелости, создании эффективной инфраструктуры как в сфере разработки ПО, так и в управлении проектами. В соответствии с моделью SEI CMM (Capability Maturity Model), в хорошо подготовленной (зрелой) организации персонал обладает технологией и инструментарием оценки качества процессов создания ПО на протяжении всего жизненного цикла ПО и на уровне всей организации.

# Конец ознакомительного фрагмента.

Текст предоставлен ООО «ЛитРес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на ЛитРес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.