

**Е. В. Дубынина**

*Заместитель начальника отдела автоматизированных систем  
управления конструкторско-технологической информацией  
Иркутский авиационный завод*

## **ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ УПРАВЛЕНИЯ ПРОЦЕССОМ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Основными задачами отдела автоматизированных систем управления конструкторско-технологической информацией (АСУ КТИ) Иркутского авиационного завода являются проектирование, разработка, внедрение и сопровождение программного обеспечения по управлению конструкторской и технологической информацией и документацией на протяжении всего жизненного цикла изделий с применением CAD / CAM / CAE / PLM технологий. С 2001 г/ отделом выполняется разработка автоматизированной системы управления документацией и данными по составу изделия (СУДП) на базе системы PLM Teamcenter.

Назначением системы является автоматизация проектно-конструкторских работ и конструкторско-технологической подготовки производства, управление массивом исходных инженерных данных, используемых для решения задач управления жизненным циклом изделий авиационной техники и их компонентов в процессе производства.

Проектирование изделия и технологическая проработка конструкторской документации выполняется в единой информационной системе управления документацией и данными по составу изделия. В настоящее время в системе работает около 2000 конструкторов и технологов подразделений завода. Основной задачей подразделения является обеспечение стабильной работы системы, своевременное выполнение разработки новой функциональности и доработки существующей по требованиям заказчиков (технического директора, главного технолога, главного конструктора).

Иркутский авиазавод имеет длительный опыт разработки программного обеспечения для информационной поддержки ключевых бизнес-процессов предприятия. Однако, в связи с постоянным увеличением сложности, повышением требований к разрабатываемому программному обеспечению, а также частой сменой потребностей пользователей и ростом требований к качеству информационных систем возникла острая необходимость усовершенствовать процесс разработки: повысить качество разрабатываемых программных продуктов, стандартизировать процесс с увеличением его эффективности.

В 2010 г. в подразделении был поставлен новый процесс разработки на основе хорошо зарекомендовавшей себя на практике методологии, которая должна быть рассчитана на разработку качественных программных систем силами разработчиков средней квалификации. Такой методологией, призванной решить перечисленные выше проблемы, и явилась методология Rational Unified Process фирмы IBM Rational Software.

Поэтому на ИАЗ было принято решение о применении принципов итерационного подхода при реализации проектов по разработке и внедрению в производство программных продуктов, связанных с доработкой функциональности PLM-системы Teamcenter по индивидуальным требованиям заказчика.

Учитывая особенности методологии RUP и особенности производства, было решено организовать процесс разработки ПО созданием внутри отдела 5 групп: анализа (автоматизации), управления требованиями, проектирования, разработки, внедрения.

Остановимся более подробно на каждом из этапов.

1. На этапе анализа, согласно методологии, выполняется разработка требований верхнего уровня и согласование с заказчиком. Создаются основы для планирования итераций и оценки стоимости и времени на разработку системы, определяются функциональные требования на ос-

нове выявленных возможностей системы. После завершения выполненных работ все наработки передаются группе управления требованиями для дальнейшей проработки. С этого момента начинается следующий этап процесса – управление требованиями.

2. На этапе управления требованиями специалистам необходимо преобразовать требования к автоматизированной системе или её частям в проектные решения, адаптировать проектные решения к среде реализации. Оформленные требования передаются далее специалистам группы проектирования.

3. Проектирование. На этом этапе специалистам после тщательной проработки требований к системе, оформленных на предыдущих этапах, необходимо выполнить проектирование архитектуры системы, проанализировать необходимость разработки вспомогательных систем и баз данных, разработать структуру кода автоматизированной системы или ее частей, организованных по уровням архитектуры, определить алгоритм работы системы. После проведения всего комплекса работ по реализации ПО на завершающем этапе необходимо выполнить интеграцию реализованной подсистемы и присоединить ее к выпуску.

4. Этап разработки включает в себя работы по непосредственному написанию исходного программного кода в среде разработки, проведению блочного тестирования разработанных компонентов и устранению дефектов.

5. При организации отдела было принято решение объединить этапы испытания и развертывания в рамках функций одной группы – внедрения, в которой было выделено две подгруппы: тестирования и технического сопровождения. Специалисты подгруппы тестирования проводят испытания модулей, разработанных по завершении этапа 4, и оформляют всю документацию, связанную с этим процессом. В обязанности персонала подгруппы технического сопровождения входит техническое и методологическое сопровождение существующих и внедряемых систем.

Таким образом, каждая итерация сама по себе выглядит как программный проект в миниатюре и включает все задачи, необходимые для выдачи мини-прироста по функциональности: планирование, анализ требований, проектирование, кодирование, тестирование и документирование. Хотя отдельная итерация, как правило, недостаточна для выпуска новой версии продукта, тем не менее, программный продукт готов к выпуску в конце каждой итерации.

На основе проведенного анализа по нормативным и фактическим затратам на разработку программного обеспечения за 2012 г. выяснилось, что в настоящее время фактическая длительность на этапах управления требованиями, тестировании и внедрении превышает плановую в 1,5 раза. Проблема срыва сроков стоит очень остро. Из-за несвоевременно выполненных работ происходит срыв сроков разработки

конструкторской и технологической документации в Управлениях главного конструктора, главного технолога и, как следствие, в производстве.

«Узким» местом в процессе является этап тестирования, так как он осуществляется почти на каждом этапе разработки ПО. Длительное тестирование требований влияет на длительность их согласования и, следовательно, на разработку ПО и эксплуатационной документации на этапе внедрения.

Еще одной проблемой, связанной с процессом разработки ПО, является обеспечение целостности и согласованности всех действий и требуемых результатов. Особенно это важно для нашего проекта, который выполняется многочисленной командой разработчиков.

Например, обычной ситуацией является изменение требований или проектных решений в процессе разработки; в этом случае должны быть каскадно изменены и приведены в соответствие все связанные, разработанные к этому времени промежуточные продукты и документы. Беда в том, что это требует высоких трудозатрат, нередко выполняется не полностью и приводит к потере качества продукта. Поэтому важно, чтобы процесс был высоко автоматизирован и поддерживался инструментальными средствами не только в части основных программных процессов, но и в отношении вспомогательных процессов, таких как конфигурационное управление, документирование и т. п. При этом важно использовать интегрируемые между собой инструментальные средства для обеспечения автоматической прослеживаемости связанных промежуточных результатов проекта. В качестве такого инструментального средства в нашем подразделении внедрен продукт IBM Rational, но не в полной функциональности, а конкретнее не автоматизирован процесс тестирования, нет связи между документацией и программным кодом.

Для выявления узких мест менеджерам и руководителю проекта необходимо контролировать длительность по каждому этапу процесса. Для отслеживания качества ПО введены показатели:

- количество итераций и замечаний на этапе Тестирования;
- количество дефектов и обращений пользователей после выполнения обновления ПО.

Для повышения эффективности разработки необходимо создать нормативно-методологическую базу и четко распределить роли исполнителей в ходе работ по проектам. Внедрение средств автоматизации IBM Rational позволит повысить качество выполнения работ и уменьшить время на принятие оперативных решений.

Для оптимизации процесса разработки ПО планируется произвести интеграцию систем контроля версий ClearCase, управления изменениями ClearQuest, календарного планирования Microsoft Project, что позволит существенно сократить затраты времени на планирование работ и организацию процесса информационного взаимодействия между группами разработчиков информационной системы.

Отдельный большой блок работ предстоит выполнить для внедрения автоматизированных средств функционального и нагрузочного тестирования. Фирмой IBM для этих целей предлагается продукт Rational Robot, но рассматриваются и другие продукты, широко известные на рынке. Преимущества, которые появляются с внедрением таких инструментов, общеизвестны. Вместе с тем для их внедрения и эффективного использования требуются значительные дополнительные ресурсы команды разработчиков. Ведь для того, чтобы протестировать основное ПО, нужно написать скрипты для автоматизации этого процесса, т. е. фактически речь идет о создании и сопровождении еще одной, вспомогательной информационной системы. На данный момент наиболее сложно решаемой представляется проблема перераспределения ресурсов разработчиков таким образом, чтобы можно было обеспечить внедрение и развитие направления автоматизированного тестирования без риска снижения темпов по развитию основной функциональности системы.

### **Список литературы**

1. Кон М. Scrum: гибкая разработка / М. Кон. – М. : Вильямс, 2011. – 576 с.
2. Кратчен Ф. Введение в Rational Unified Process / Ф. Кратчен. – М. : Вильямс, 2002. – 240с.
3. Методика управления качеством. IBM Rational ClearCase. – М. : СМ-Консалт, 2007. – 93 с.
4. Методика управления качеством. IBM Rational ClearQuest. – М. : СМ-Консалт, 2007. – 61 с.
5. Руководство по RUP. – М. : Кудиц-Образ, 2004. – 432 с.
6. Software Factories: потоковая сборка типовых приложений, моделирование, структуры и инструменты. – М. : Диалектика, 2006. –592 с.