

Инженерное *программирование* – это планирование и *управление проектом*, распределения всех видов ресурсов – исполнителей, программных и инструментальных средств и *оценивание* сроков, стоимости *работ* для успешного выполнения проекта. С проектом связано также выявление рисков и других обстоятельств, замедляющих изготовления версий (конфигурации) системы [11.1]–[11.6][11.7].

Материал данной темы состоит из трех разделов, связанных с проблемой управления проектом:

1. Методы управления программным проектом.
2. Методы обнаружения и устранения рисков в проекте.
3. Управление конфигурацией системы.

11.1. Методы управления проектами

Согласно мировой статистики, не все реализуемые программные проекты завершаются успешно, 33% из них являются провальными по следующим причинам:

- требования заказчика не выполняются,
- проект не вложился в стоимость или сроки,
- этапы и виды работ оказались несоординированными друг с другом,
- менеджер не ориентировал разработчиков проекта на применение новейших методов и средств программирования,
- не проводилось должного планирования и соблюдения стандартных соглашений по применению процессов ЖЦ.

Основные положения управления проектом, задачи и методы которого отработывались на технических проектах (например, первый проект разработки лайнера для перевозки пассажиров из Европы в Америку), привели к созданию Генри Гантом диаграммной схемы для учета времени выполнения проекта. Эта схема прошла многолетнее развитие и на данный момент задачи управления проектом сформулированы следующим образом:

- планирование проекта и составление графиков работ выполнения проекта,
- управление проектными работами и командой исполнителей,
- управление рисками,– оценивание продукта и процессов в целях их дальнейшего усовершенствования и др.

Управление проектом – это руководство работами команды исполнителей проекта для реализации проекта с использованием методов управления, планирования и контроля *работ, управление рисками*, эффективной организацией работы и коммуникационными потоками в команде исполнителей.

Основными составляющими любого проекта являются следующие: **ресурсы** (людские, финансовые и технические), **время и стоимость** выполнения проекта. Ответственность за координацию и реализацию этих трех составляющих несет *менеджер проекта*, а ответственность за идейную, функциональную сторону проекта несет главный программист.

Успешное выполнение проекта зависит от знания менеджером методов управления проекта и особенностей программного продукта:

- продукт не материален, его нельзя увидеть, пощупать в процессе конструирования (как это имеет место, например, при строительстве здания) и повлиять оперативно на его структуру, управление конфигурацией и реализацию;
- стандарты ЖЦ должны быть адаптированы на нужный вид и тип продукта, как это имеет место в технических дисциплинах (автомобильной, авиационной и др.), и разработать методики выполнения проектных решений исполнителями;
- программные продукты создают длительное время на компьютерной технике, которая быстро устаревает и обновляется ее архитектура;
- появляются новые и новые языки программирования.

Накопленный *опыт* в создании технических проектов был систематизирован (в Институте управления проектами в США) и разработано *ядро знаний* – PMBOK (*Project Management Body of Knowledge* [11.2]. В нем малыми проектами считаются проекты, содержащие 100 *работ* и 15 исполнителей, средними – 500 *работ* и 50 исполнителей и большими – 1000 *работ* и 100 исполнителей.

В ядре PMBOK определены основные аспекты разработки проектов:

методы управления, планирования и контроля работ;

эффективная организация проектной команды;

инструментарий менеджера проекта (например, система Project Management фирмы Microsoft).

11.1.1 Методы управления программным проектом

К настоящему времени сформировалось несколько методов, эффективно применяемых в практике реализации программных проектов [11.5, 11.7]. Рассмотрим их.

Метод критического пути CPM. Основополагающий момент в создании этого метода – исследование возможности эффективного использования вычислительной машины Univac на фирме "Dupon" при планировании и создании планов-графиков больших комплексов работ по модернизации заводов этой фирмы. В результате был создан рациональный и простой метод (Уолкера-Келли) управления проектом с использованием ЭВМ, который был назван *CPM (Critical Path Method)* – метод критического пути.

Критический путь – наиболее полный путь работ в сетевом графике, которые лежат на этом пути. Именно продолжительность критического пути определяет наименьшую общую продолжительность работ в проекте в целом. Время выполнения всего проекта может быть сокращено за счет уменьшения времени выполнения задач, которые лежат на критическом пути. Соответственно любая задержка выполнения задач критического пути приводит к увеличению времени выполнения проекта. Эта концепция обеспечивает концентрацию внимания менеджера на критических работах. Однако основное преимущество метода критического пути – управление сроками выполнения задач, которые не лежат на критическом пути. Этот метод позволяет рассчитать возможные календарные графики выполнения комплекса работ на основе описанной логической структуры сети и оценок времени выполнения каждой работы [11.1–11.7].

Метод основан на графическом представлении задач (работ) и видов действий на проекте и задании ориентировочного времени их выполнения в виде графа (рис. 11.1), в вершинах которого располагаются работы и время выполнения каждой работы под вершинами либо на дугах графа.

Граф целесообразно строить тогда, когда работы и время их выполнения являются определенными. Критический путь в графе указывает максимальную продолжительность работ на графе (от начальной работы до последней).

При выполнении проекта выбираются и выполняются работы, которые не влияют на время выполнения других (независимых) работ проекта или на их продолжительность. Работы на критическом пути могут сокращаться за счет изменения времени выполнения.

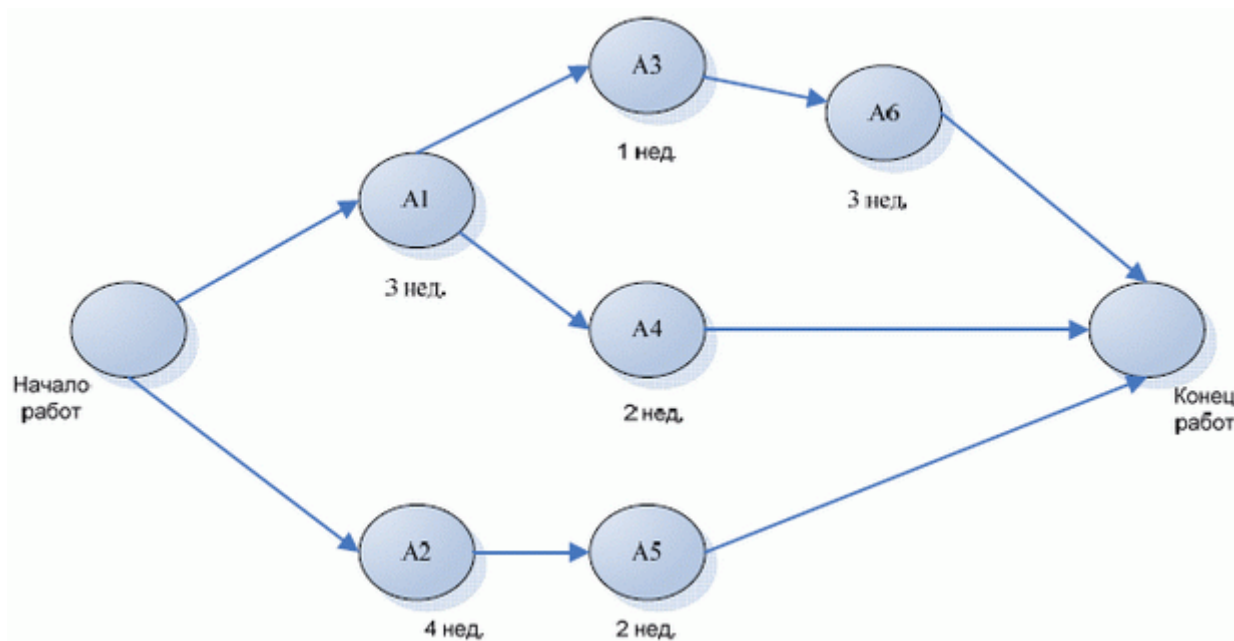


Рис. 11.1. Граф задания сроков выполнения работ

Представленный в таком виде график работ называется *сетевой диаграммой* и служит для графического отображения работ проекта, их взаимосвязей, последовательностей и времени выполнения. В графе вершины отображают работы, а линии – взаимные связи между работами. Этот граф – наиболее распространенный способ представления сети на сегодняшний день.

Метод анализа и оценки PERT. Параллельно с разработкой *CPM* в военноморских силах США был создан (фирма "Буз, Аллен & Гамильтон") метод анализа и оценки программ *PERT (Program Evaluation and Review Technique)* для реализации проекта разработки ракетной системы "Polaris", объединяющей около 3800 подрядчиков с числом операций более 60 тыс. [11.6].

Применение метода *PERT* позволяло руководству данной программы точно знать, что нужно делать в каждый момент времени и какой исполнитель эту работу выполняет, а также определять вероятность своевременного завершения отдельных операций на процессах проекта. Руководство программой создания ракетной системы по методу *PERT* оказалось настолько успешным, что проект удалось завершить на два года раньше запланированного срока.

Метод *PERT* представляется *сетевыми диаграммами* с вершинами-событиями, а работа – в виде линии между двумя событиями, отображающими начало и конец работы. В целом расхождения между этими двумя методами сетевого представления графа работ – незначительные. Однако этот метод, в отличие от СМР, учитывает возникающие неопределенности во времени выполнения каждой операции.

Представление более сложных связей между работами для задания узлов графа в виде вершина-событие является более сложным, и потому этот метод реже используется на практике.

Возможное время выполнения операций оценивается с помощью трех оценок:

оптимистичной (O);
 пессимистической (P);
 вероятностной (B).

$$\frac{O+4B+P}{6}$$

Далее возможное время вычисляется по формуле:

Для уменьшения провалов в проекте используются и другие аналитические методы и усовершенствованные технологические системы разработки. Как показывает опыт, "продвинутые" технологии предлагают следующие постулаты успешного создания проекта [11.3]:

1. Проект начинать с правильного шага;
 2. Поддержка темпа работы на проекте;
 3. Обеспечение прогресса и принятия правильных решений;
 4. Анализ заверщенного проекта (преимуществ и недостатков).
1. **Проект начинать с правильного шага.** К правильному шагу относится формирование команды разработчиков из хороших специалистов, в том числе не более 20% звезд, наиболее приспособленных для хорошей работы над проектом (много звезд – создание конфликтов). В команду входят надежные разработчики с совместимыми характерами и рабочими привычками. Звезды решают сложные вопросы, разрабатывают более ответственные алгоритмы и проводят техническое обучение остальных членов команды. Для уточнения всех особенностей проекта разработчики садятся за стол переговоров с заказчиком и составляют взаимно приемлемый документ.
 2. **Поддержка темпа работы** предполагает:
 - уменьшение текучести кадров;
 - контроль качества выполняемых работ;
 - управление процессом разработки продукта, а не людьми команды.

Текучесть кадров – проблема программной индустрии, так как перемещение или уход отдельного специалиста вынуждают других членов группы к трудоемкой работе, связанной с изучением незаконченных и не полностью документированных программ. Большой промежуток времени между уходом специалиста и нахождением замены может привести к краху отдельных частей проекта.

Контроль качества – необходимое условие создания качественного проекта. Им занимаются с самого начала разработки проекта, устанавливая процедуры проверки качества, используя при этом разработчиков, которые могут создавать высококачественный продукт.

Управление процессом разработки состоит в анализе проектирования элементов системы и критике результатов труда исполнителей, а не режима работы.
 3. **Поддержка прогресса и правильных решений.** Программа отличается от других продуктов тем, что она – неосвязаема. Ее разработка начинается с создания концептуальной модели, а результаты ее применения влияют на получение продукта.
 4. **Анализ проекта.** Это изучение своих ошибок при реализации предыдущего проекта, чтобы не повторить их в новом проекте. Так как каждая команда и фирма имеют свои особенности, которые влияют на процесс разработки, то при анализе можно выделить особенности и закономерности, которые влияют на получение качественных результатов.

11.1.2. Планирование проекта

Планирование – это процесс распределения и назначения ресурсов (материальных и людских) с учетом стоимости и времени выполнения проекта, является необходимой предпосылкой выполнения любой, даже самой простой задачи. Неадекватное планирование может привести к срыву проекта или к получению неадекватных результатов в проекте.

Планирование и перепланирование – наиболее емкие во времени части управления проектом, в особенности на ранних стадиях проекта. В прошлом проекты в области программных систем не имели планов и оценок их следования и выполнения. Современные методики предлагают средства для исправления этой ситуации, предоставляя в распоряжение менеджеров инструменты и методы, которые разрешают работать в направлении достижения реальных и достижимых оценок работ и планов. Планирование в том или ином виде выполняется в течение всего срока реализации проекта. В начале проекта обычно разрабатывается неофициальный первоначальный план – грубое представление о том, что потребуется выполнить в случае реализации проекта. Решение о выборе проекта в значительной мере базируется на оценках этого плана. Формальное и детальное планирование начинается после принятия решения о реализации проекта.

Планирование заключается в составлении планов:

- работ со сроками их выполнения по методу критического пути CPM или PERT;
- обеспечении требуемого качества и контроля промежуточных результатов процессов ЖЦ;
- управления рисками;
- аттестации результатов проектирования и деятельности исполнителей проекта;
- управления конфигурацией и др.

Составляется график работ по следующей схеме (рис. 11.2):



Рис. 11.2. Шаги составления графика работ на проекте

При планировании по методу PERT событие или дата в плане – это некоторая веха осуществления отдельных работ проекта, она используется для отображения (отметки) состояния завершения тех или иных работ. В контексте проекта менеджеры используют вехи для того, чтобы обозначить важные промежуточные результаты, которые должны быть достигнуты в процессе реализации проекта. Последовательность вех, определенных менеджером, часто называется *планом по вехам* или *по событиям*. Определение плана достижения соответствующих вех образует календарный план на основе вех.

На этапе планирования могут использоваться также сетевая разбивка работ (СРР) и *диаграммы Ганта* [11.7]. СРР – это иерархическая структура декомпозиции задач проекта на подзадачи. На нижнем уровне располагаются работы, детализированные на элементы деятельности в сетевой модели СРР.

Они отображают:

- структуризацию работ на основные компоненты и подкомпоненты;
- определение направлений деятельности для достижения комплексных целей;
- распределение ответственных за выполнение отдельных работ на проекте;
- оформление отчетности с обобщением информации по результатам проекту.

План отражает этапы, состояние работ на проекте и разделение каждой работы в виде отдельных задач, а также связи между процессами, интервалы времени выполнения каждой деятельности, время начала и завершения работ. В план работ включаются разные виды демонстраций: функций, подсистем, надежности, средств защиты и др. К документам плана относятся: комплект руководств и методик для выполнения заданных операций, связей системы с другими подсистемами и др.

План в виде графа СРР имеет этапы, шаги и деятельности, а также начало и конечную деятельность на процессе (рис. 11.3).

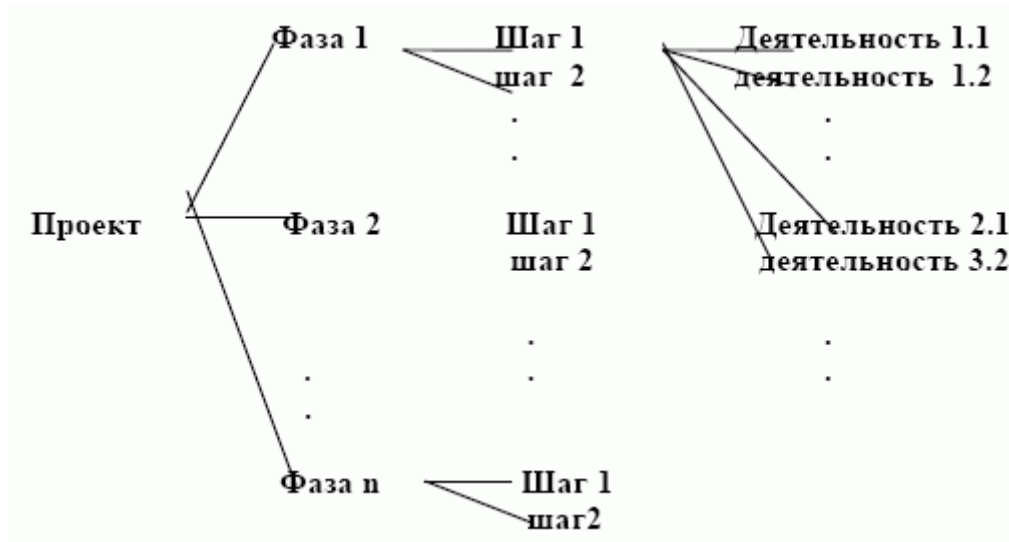


Рис. 11.3. Пошаговый граф плана проекта

Диаграмма Ганта – это линейная диаграмма, на которой задачи проекта представляются сроками в виде отрезков времени, включают даты начала и окончания выполнения задач с учетом возможных задержек или других временных параметров. Другой формой представления – сетевой график работ, задаваемый в виде графа, в вершинах которого располагаются пункты работ плана, а на дугах количество дней (недель) для их выполнения (рис. 11.4).

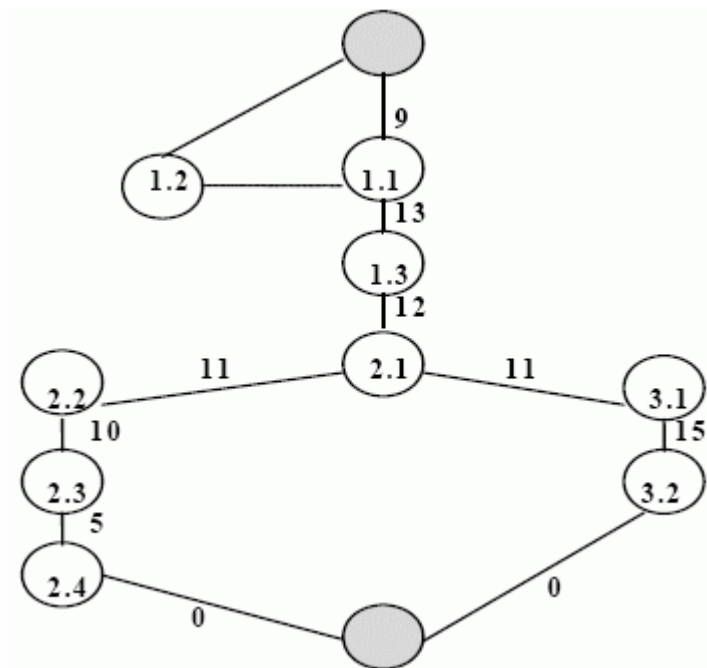


Рис. 11.4. Вид графа работ и сроков (на дугах) для проекта

Управление процессами на проекте состоит в определении:

начальной точки – события или набора событий, которые произошли до начала выполнения этапа процесса и для которого описывается набор условий, включая начало процесса;

продолжительности – интервала времени, за который процесс должен успешно завершить свое выполнение;

срока – даты, до которой процесс полностью или частично завершает свое выполнение;

конечной точки процесса – контрольной точки, в которой заказчик проверяет качество полученных результатов процесса.

Дуге, выходящей из начальной вершины и входящей в заключительную вершину, соответствует временная отметка 0. С помощью этих меток задается время выполнения процесса.

В графе могут присутствовать циклические пути. По графу проводят анализ критических путей, т.е. определяют данные о продолжительности каждого процесса. План проекта проводится в терминах этапов: планирование, проектирование, кодирование, тестирование и сопровождение. Для первых двух методов

планирование затрагивает: определение спецификаций, бюджета и расписания, а также развития плана проекта в целом.

Современные средства управления проектами обеспечивают визуализацию структуры графа проекта и процессов выполнения работ. Это дает возможность видеть виды деятельности, которые могут выполняться последовательно или параллельно или они находятся на критическом пути (например, в системе Project Management).

11.1.3. Организационные аспекты управления проектом

Распределение работ по исполнителям. Наиболее часто определение ролей исполнителей проекта соответствует этапам разработки, которые отображаются на графе. По графу проводится анализ критических путей и определение данных о продолжительности каждого процесса. План проекта реализуется с помощью методов планирования, проектирования, кодирования, тестирования и сопровождения. При этом планирование затрагивает бюджет и план развития проекта в целом.

Состав и количество сотрудников, входящих в команду проекта, зависит от масштаба работ и опыта сотрудников. Сотрудники должны быть квалифицированными, способными выявить ошибки и неточности в проекте на самых ранних стадиях ведения проекта. Разделение труда сотрудников по этапам имеет свои определенные преимущества, но требует специальной техники общения между группами сотрудников для эффективной работы (проверки, просмотры, откаты назад, сквозной контроль).

Специалисты, которые наиболее подходят к выполнению каждой из перечисленных ролей, различаются между собой:

- способностью выполнять работу;
- интересом к работе;
- опытом работы с подобным проектом, инструментами, языками, технологиями и ОС;
- способностью к обучению;
- коммуникабельностью с другими сотрудниками;
- способностью разделить ответственность с другими;
- профессионализмом и знанием методов управления. Менеджер проекта должен учитывать способности сотрудников выполнить определенные работы по проектированию или по тестированию системы в целом в заданный срок. Работающие в одной группе исполнители должны разделять одни и те же взгляды по проведению порученной им работы и пользоваться одним стилем программирования. Разделение большого участка работы на меньшие части должно соответствовать фрагментам работы на этапе, определению ролей и ответственности каждого сотрудника в проекте.

Организационная структура проекта. Для хорошей организации ведения проекта подбирается подходящая структура проекта на основании следующих данных:

- рабочие стили членов группы;
- число людей в группе;
- стиль работы с заказчиками и разработчиками.

Один из популярных стилей ведения проекта впервые использовался в IBM (рис. 11.5). В нем главным ответственным за проектирование системы и ведение разработки является руководитель группы программистов. Ему непосредственно подчиняются программисты, которые имеют право последнего слова при принятии решений – главные программисты.

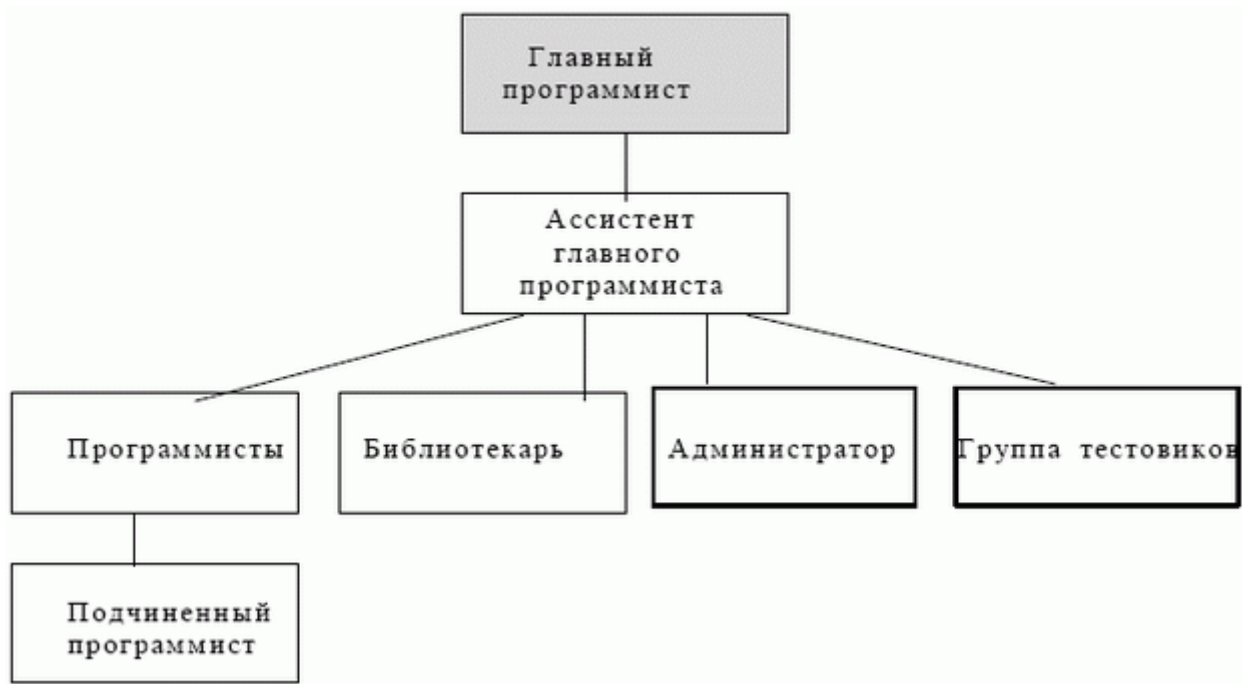


Рис. 11.5. Структура организации группы главного программиста

Главный программист руководит своей подгруппой программистов и непосредственно посвящен в детали проекта и разработки программы. Ассистент главного программиста дублирует, замещает главного программиста, когда это необходимо. Библиотекарь отвечает за всю документацию проекта: компилирование и тестирование всех модулей библиотеки. В группу входит администратор и группа тестовиков. Старшие программисты и младшие непосредственно подчиняются старшим.

Хотя структура такой рабочей группы иерархическая, каждый член группы может общаться непосредственно с главным программистом или с другими сотрудниками. Главный программист должен сам просматривать основные решения на проекте.

Альтернативная структура ведения проекта описана Вейнбергом (Weinberg) [11.3] – обезличенное программирование, при котором все несут одинаковую ответственность за качество продукта. Такая структура подходит для маленьких групп программистов.

Ответственность за моделирование работ в проекте. В [11.3] в рамках военного ведомства разработана общая структура команды для создания интегрированного продукта (*Integrated Product Development Team*). Соответствующая модель приведена на рис. 11.6.

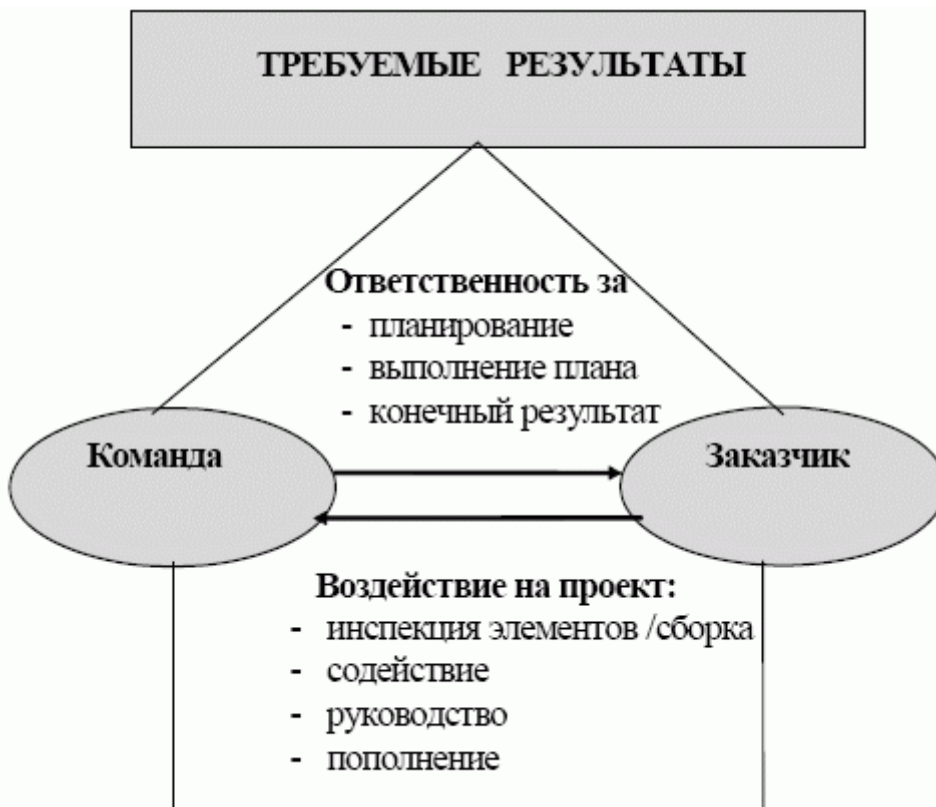


Рис. 11.6. Модель ответственности лиц в интегрированном проекте

Программирование элементов системы осуществлялось с помощью компилятора Ада на этапах ЖЦ: проектирование, анализ и построение ПО, управление конфигурацией, тестирование, объединение элементов проекта с помощью интерфейса для перехода от фреймов к рабочим станциям. Менеджер проекта управлял *распределением обязанностей* и сроками выполнения трех одинаковых по размеру задач проекта.

Участники проекта работали по *матричной организации*, при которой каждый инженер выполнял определенный тип работ (проектирование или тестирование) в одном или более проектах. Суть организации рабочей группы интегрированного создания продукта состоит в возможности работать сообща в соответствии с общей дисциплиной для всех типов групп и средствами выполнения.

Группа – это комбинация разных сотрудников, в которой каждый ответственный за результат своей работы. Заказчик влияет на результат или на выбор пути для достижения результата. В процессе разработки обязанности и *роли сотрудников* могут меняться. В планах устанавливаются сроки в пределах недели или часов. Для организации работ большого коллектива используются карты обязанностей, схемы со сроками выполнения работ для каждой части проекта. Выполненное задание отмечается в диаграмме планируемых и реально выполненных работ. Модель обязанностей предполагает использование сценариев и шаблонов взаимодействия между сотрудниками, при которых результат работы одной группы передается как исходное данное для работы другой группе.

11.1.4. Системы управления проектом

MS Project 2003. Для целей управления проектом корпорация Microsoft предлагает систему управления проектом MS Project 2003, которой пользуются более 8 миллионов физических и юридических лиц для планирования работ малых и средних организаций [11.8–11.12, 11.15]. Эта система включает в себя набор базовых функций, большое количество специальных функций, свойственных пакетам профессионального уровня. Базовые функции на стадии планирования позволяют ответить на вопросы, связанные с возможностью реализации проекта, оценки его стоимости и распределения финансовых затрат на работы в проекте, а также осуществить подбор состава исполнителей для реализации проекта и др.

В системе проводится полный анализ проекта по методу СРМ с планированием, но без детализации и с использованием шаблонов, которые входят в состав пакета. Каждый шаблон относится к определенной области деятельности и может считаться некоторым стандартным планом проекта. Внося необходимые поправки, можно получить реальную оценку нужных затрат. Если соответствующий шаблон отсутствует, то структура проекта создается вручную.

При этом MS Project оказывает помощь в построении сетевого графика путем задания параметров (длительность проекта, календарные даты начала и окончания работ и т.д.). На основе сетевого графика автоматически формируется календарный план в виде *диаграммы Ганта* со сроками, которые относятся к реальным датам.

Для оценки состава исполнителей менеджер вводит в систему представление о характере работ на проекте, сроках выполнения работ и характеристике исполнителей. На основе этих данных и видах ресурсов в проекте строится гистограмма загрузки и стоимости работ. Для учета рисков в системе (ресурсных, человеческих и др.) в составе MS Project имеется процедура анализа рисков проекта с помощью метода PERT, который дает оценку возможности возникновения рисков в проекте.

Можно применять дополнительные методы и средства, выбор которых зависит от специфики проекта и уровня подготовки пользователя. Например, простым и вместе с тем эффективным средством является сравнение нескольких версий (сценариев) оценок проекта. Как правило, таких оценок может быть три: пессимистическая, оптимистичная и наиболее вероятностная.

Стандартизация процесса управления проектом. Процесс управления проектом входит в ЖЦ стандартов ISO/IEC 12207-2002 и ISO 15504 (части 1-9) 2002. В первом стандарте он представлен как самостоятельный процесс и вошел в организационную категорию процессов ЖЦ. Этот процесс рекомендует создавать структурную группу для планирования и управления проектом, учитывающую задачи и действия, выполняемые на процессах ЖЦ. Согласно стандарту ISO15504 процесс управления проектом состоит в идентификации, координации и контроле действий, задач и ресурсов, используемых для изготовления продукта и/или услуги в соответствии с заданными требованиями к проекту.

Главные факторы осуществления задач программного проекта:

- объем работ, имеющиеся ресурсы и ограничения;
- стоимость выполнения задач и необходимых ресурсов;
- компетентность специалистов;
- методы и средства и стандарты, соответствующие выполнению задач проекта;- планы проекта и контроль их выполнения;
- оценка стоимости, трудоемкости и продолжительности выполнения проекта;
- измерение размера и сложности продуктов, полученных на процессах ЖЦ.

Методы и средства выполнения процесса управления проектом ориентированы на методическую и инструментальнотехнологическую поддержку процессов ЖЦ при разработке проекта.

Для реализации проектов разработаны специализированные программные средства *Adaptable Process Model*, а также методы PERT или CPM, позволяющие спланировать работы проекта, распределить время и специалистов по работам. Руководитель плана управляет выполнением плана, собирает данные и осуществляет статистическую оценку сделанных работ и достижение характеристик проекта (например, заданную длительность выполнения процесса). Более глубокий анализ характеристик проекта и процесса его выполнения обеспечивает применение методов формального моделирования процессов (конечные автоматы, сети Петри, байесовские сети и т.п.).

11.1.5. Оценивание стоимости проекта

Одна из наиболее важных работ на проекте – оценка стоимости проекта, которая определяется исходя из стоимости отдельных частей проекта, условий выполнения работ, фактического штата исполнителей, используемых методов и инструментов. В стоимость проекта входит все, что создает стиль ведения проекта: компьютеры, ПО, площади, мебель, телефоны, модемы и многое другое. Иногда должны быть созданы дополнительные условия (например, безопасность).

К дополнительным расходам на проекте относятся приобретение системы тестирования, CASE системы и др. Главная оценка в проекте – это оценка затрат на ведение проекта, выражаемая в человекоднях работ исполнителей на проекте. Эти оценки проводятся на ранней стадии ведения и составления плана. Общая стоимость проекта оценивается опытными специалистами с погрешностью до 10%. Метод оценки может быть "сверху-вниз", "снизу-вверх" или исходить из стоимости ранее изготовленного проекта. Эксперты проводят пессимистическую, оптимистическую и реальную оценку путем опроса всех членов рабочей группы и коррекции каждой оценки для получения наиболее правдоподобной. В некоторых рабочих группах пессимистическая и оптимистическая оценки могут сильно отличаться. Методы оценки работ имеют недостатки, связанные отличиями модуля старой системы от новой.

Алгоритмические методы оценки. Они отображают связи между затратами в проекте и факторами, которые на них влияют. Стоимость проекта можно определить по формуле:

$$E = (a + bS^c)m(X),$$

где S – оценка размера системы, a, b, c – эмпирические константы, $X[n]$ – вектор факторов стоимости, m – регулирующий множитель, основанный на затратах факторах.

В [11.3] предложена модель оценки, полученная экспериментальным путем: $E = 5.25S^{0.91}$. Эта модель применялась при оценке проекта, в котором программы имели размер от 4 000 до 467 000 строк

кода и были написаны на 28 различных ЯП для 66 компьютеров. На разработку было затрачено от 12 до 11758 человекомесяцев.

В [11.4] предложено уравнение затрат организации-разработчика:.

$$E = 5.5 + 0.73S^{1.16}$$

Модель *COSOMO* Боема [11.3] объединила в себе три техники оценки проекта. В первой модели использовались показатели цены, персонал и свойства проекта, продукта и среды. Модель основывалась на трех стадиях ведения проекта. *На первой стадии* строится прототип системы для задач повышенного риска и проводится оценка затрат (например, число таблиц в БД, экраны и отчетные формы др.). *На второй стадии* проводится оценка затрат на проектирование и реализацию функциональных точек проекта согласно требований к проекту.

На третьей стадии проводится оценка размера системы в готовых строках программ и других факторов.

Для оценки использовалось следующее уравнение: $E = bS^c m(X)$, где первичная оценка bS^c корректируется с помощью вектора стоимости $m(X)$ и учета числа старых и новых объектов. Параметр c изменяется от 0 до 1.0 для первой стадии и от 1.01 до 1.26 для остальных стадий.

В качестве итога приведем основные моменты эффективного и успешного управления программным проектом [11.3, 11.13]:

1. Определение границ системы и точек выполнения разработки. Формирование временного плана выполнения работ в точках проекта.
2. Определение структуры рабочей группы, а также видов работ и ресурсов.
3. Техническое описание состава планируемой системы (ПО, компиляторы, интерфейсы, оборудование и др.) и ограничений на время, производительность и т.п.
4. Использование стандартов, процедур, техник и инструментов ведения проекта.
5. Разработка планов достижения качества, управления конфигурацией и документацией.
6. Разработка плана управления данными и источниками информации.
7. Разработка плана тестирования, измерения и оценивания результатов работ.
8. Составление плана обучения пользователей системы.
9. Определение плана обеспечения безопасности проекта (конфиденциальность, пароли и др.).
10. Составление плана управления рисками.
11. Разработка плана сопровождения, в котором указаны ответственные за изменение кода, ремонт оборудования, изменение документации и др.
12. Описание алгоритмов, инструментов, техники просмотра или инспекции кода, языков ведения проекта, языков кодирования и тестирования.

11.2. Методы управления рисками в проекте

Причиной возникновения рисков являются неопределенности, существующие в каждом проекте. Риски могут быть "известные", которые можно планировать, "неизвестные", которые не идентифицированы и не могут быть спрогнозированы [11.9-11.15].

Риск – это нежелательное событие, которое может иметь непредвиденные негативные последствия. Если в проекте идентифицировано множество возможных событий риска, которые могут повлечь за собой негативные последствия, то такой проект является склонным к риску.

Многие компании уделяют внимание разработке и применению корпоративных методов управления рисками, которые учитывают специфику проектов и методы управления. Американским Институтом управления проектами (*Project Management Institute*) разработан стандарт по управлению проектами PMBOK. В нем переработаны разделы и предложены следующие процедуры управления рисками:

1. Планирование управления рисками;
2. Идентификация рисков, способных повлиять на проект, документирование их характеристик;
3. Качественный анализ рисков и условий их возникновения с целью определения их влияния на успех проекта;
4. Количественный анализ вероятности возникновения и влияния *последствий рисков* на проект;
5. Планирование процедур и методов по ослаблению отрицательных последствий рисков событий;
6. Мониторинг и контроль рисков, оценка действий по минимизации рисков в проекте;

Все эти процедуры связаны друг с другом, каждая из них выполняется, по крайней мере, один раз в проекте. Их можно рассматривать как дискретные элементы с четко определенными характеристиками.

Планирование управления рисками – это процесс *принятия решений* по применению и планированию управления рисками для конкретного проекта. Этот процесс может включать в себя решения по организации, кадровому обеспечению процедур управления рисками, выбор предпочтительной методологии, данных для идентификации риска, временного интервала для анализа ситуации. Планирование управлением риска адекватно уровню и типу риска, а также важности проекта в организации.

Идентификация рисков выполняется для определения рисков, которые способны повлиять на проект. *Идентификация* рисков должна проводиться регулярно на протяжении реализации проекта. К вопросу идентификации должны привлекаться менеджеры проекта, заказчики, пользователи и независимые специалисты. Эта процедура выполняется как итерационный процесс. Первоначально *идентификация* рисков может быть выполнена отдельными менеджерами проекта или группой аналитиков рисков. Далее идентификацией может заниматься основная *группа* исполнителей проекта. Для формирования объективной оценки на завершающей стадии могут также участвовать независимые специалисты.

Качественная оценка рисков – это процесс качественного анализа идентификации рисков и *оценка риска*, требующего быстрого реагирования. Такая оценка определяет степень важности риска и способа реагирования. Расстановка приоритетов для разных категорий рисков обеспечивается доступностью информации на проекте. Качественная оценка рисков включает в себя оценку условий возникновения рисков и определения их воздействия на проект с помощью стандартных методов и средств, применение которых помогает частично избежать неопределенности, которые часто встречаются в проекте. В течение ЖЦ проекта должна происходить постоянная переоценка рисков.

Количественная оценка рисков определяет *вероятность* возникновения рисков, влияние *последствий рисков* на проект и принятие решений по оценке риска:

- оценка вероятности достижения конечной цели проекта;
- определение степени воздействия риска на проект и оценка объемов непредвиденных затрат и материалов, которые могут понадобиться;
- скорейшее реагирование на риск и определение влияния его последствий на проект;
- оценка фактических затрат и определение предполагаемых сроков окончания работ на проекте.

Количественная оценка рисков включает и качественную оценку и основывается на идентификации рисков. Количественная и качественная оценки рисков могут применяться в отдельности или вместе в зависимости от заданных времени и бюджета для проекта.

Планирование реакций на риски – это разработка методов и технологий снижения отрицательного воздействия рисков на проект. Планирование включает в себя идентификацию и распределение каждого риска по категориям и служит способом защиты проекта от воздействия рисков на проект. Эффективность своевременной плановой реакции на риск определяет результат проекта (положительный или отрицательный).

Стратегия планирования реакции на риск должна соответствовать типам рисков, рентабельности ресурсов и временным параметрам предотвращения рисков. Вариантов стратегии реакции на риски может быть несколько. При этом риски должны быть адекватны задачам на каждой стадии проекта и согласованы со всеми членами группы по управлению проектом.

Мониторинг и контроль – это процедуры слежения за идентификацией рисков, обеспечению выполнения плана рисков и оценки его эффективности с учетом понижения риска. Показатели рисков, связанные с осуществлением условий выполнения плана, фиксируются. *Мониторинг* и *контроль* сопровождает процесс внедрения проекта в жизнь. Качественный *контроль* выполнения проекта является источником информации, помогающей принимать эффективные решения для предотвращения возникающих рисков. Цель мониторинга и контроля состоит в выяснении таких ситуаций:

- реакция на риски внедрена в соответствии с планом и необходимыми изменениями,
- оценка изменения рисков по сравнению с предыдущим риском,
- определение влияния риска на принятие необходимых мер,
- реакция на риски должна быть запланированной или не быть случайным процессом.

Контроль может повлечь за собой выбор альтернативных стратегий, принятие некоторых коррективов, перепланировку проекта для достижения базового плана. Между менеджерами проекта и группой риска должно быть постоянное взаимодействие и фиксация всех изменений. Отчеты по выполнению проекта и системе рисков постоянно пополняются.

Управление рисками основывается на рассмотрении двух основных типов рисков: *общего риска* для всех типов проектов и специфического риска.

К первому типу риска относится риск, возникающий при непонимании требований, при нехватке профессионалов или недостатке времени на тестирование. Риск второго типа выражается недостатками проекта (незавершенность проекта к обещанному сроку и др.). Управление риском включает в себя шаги управления риском (см. [рис. 11.7](#)).

Для каждого возможного риска определяется показатель степени его вероятности и потерь, связанных с риском. Во время проведения *регрессионного тестирования* отыскиваются критические ошибки.

В зависимости от того, насколько эта ошибка критична, и от того, какие показатели риска действуют, вычисляется *ущерб* риска.

Деятельность по управлению риском включает выполнение задач: *уменьшения риска*, планирования риска, решения на исправление обнаруженного риска. *Уменьшение риска* достигается за счет избегания риска при изменении требований, перераспределения риска, отслеживания риска и управления.



Рис. 11.7. Шаги по управлению риском

Систему управления риском можно представить в виде отношения:

Ущерб до минимизации - ущерб после минимизации

Цена минимизации риска

Минимизация риска может быть получена путем прототипирования [11.5] и рассмотрения возникающих *причин* риска в проекте: сокращение штата или набор неквалифицированных сотрудников, неудачная постановка или изменение требований, неумение работать в реальном времени, ограниченные компьютерные ресурсы и др.

11.3. Управление конфигурацией программной системы

Под *конфигурацией системы* понимается конкретная версия ПС для разных ОС, компьютеров и включает в себя функции, объединенные между собой процедурами связи (или развертывания) и параметрами, которые задают режимы функционирования системы в среде ОС [11.14]. Выпуск версии системы выполняется в целях

ее поставки заказчику. Процесс получения конкретной версии системы можно представить в виде схемы (рис. 11.8).

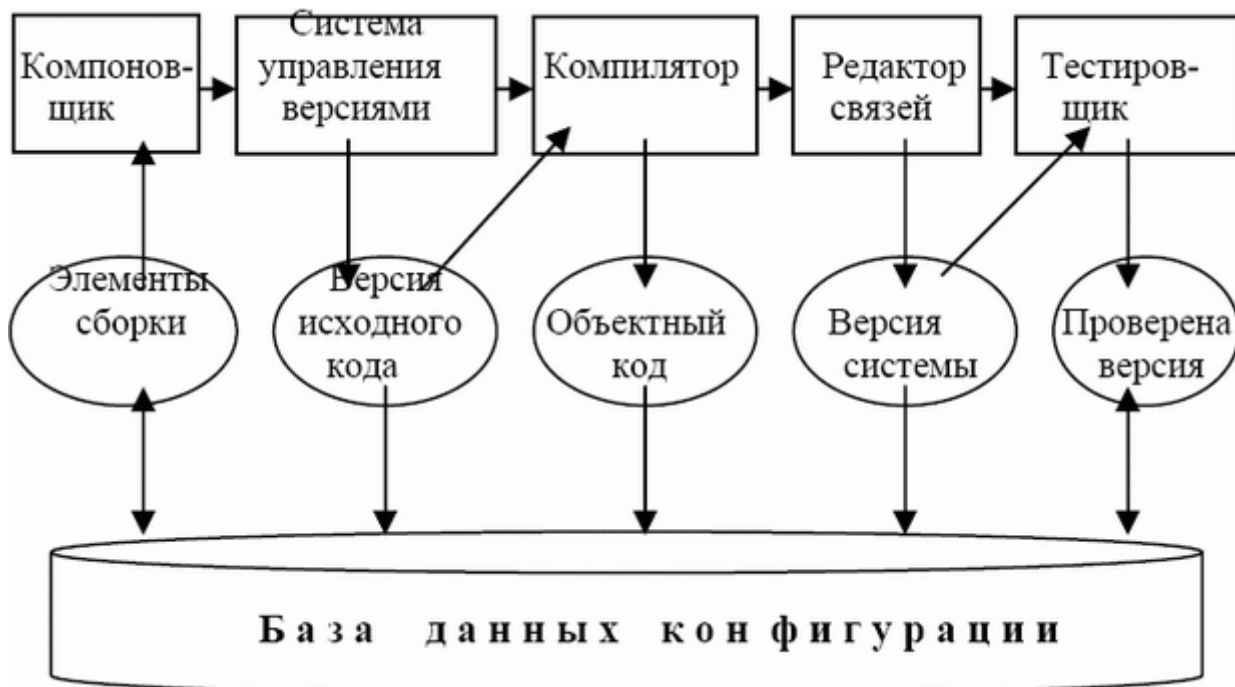


Рис. 11.8. Схема формирования версии системы

Выбираются компоненты системы из базы данных конфигурации, реализующие заданные функции, которые собираются компоновщиком системы, а система управления версиями создает исходную версию для компиляции и получения объектного кода, редактируемого в готовую версию.

Изготовленная версия системы проверяется на все виды связей, последовательное выполнение компонентов и получение контрольного результата.

Среди компонентов сборки могут быть такие, которые изменялись и отдельно проверялись. Поэтому изготовленная версия системы должна обязательно тестироваться на тестовых наборах данных, а результаты сравниваться с ожидаемыми результатами.

Версия или конфигурация системы состоит из:

- базиса конфигурации (*Configuration Baseline*) – формально созданной основы (версии) системы из отдельных компонентов и документации, позволяющей проводить дальнейшее развитие системы;
- элементов конфигурации (*Configuration Item*), выделенных для управления или обработки функций системы на процессорах компьютеров системы;
- программных компонентов, выполняющих задачи в сформированной версии системы.

Базис конфигурации определяет технические решения, перечень главных элементов конфигурации, значения параметров и специализированные процедуры их связи и развертывания версии системы из компонентов в заданной последовательности. Чем больше в системе компонентов, тем больше вероятность того, что некоторые из них могут иметь ошибки. Это приводит к необходимости исправления обнаруженных ошибок, уточнений или дополнений, как новых функций, так и технических средств (компьютеров, оборудования и др.).

Для каждого возможного риска определяется показатель степени его вероятности и потерь, связанных с риском. Во время проведения регрессионного тестирования отыскиваются критические ошибки.

11.3.1. Управление и планирование конфигурацией

Управление конфигурацией – процесс, обеспечивающий идентификацию элементов конфигурации системы при ее создании для проведения систематического контроля, учета и аудита внесенных изменений, а также для поддержки целостности и работоспособности системы. Согласно стандарту IEEE Std.610-90 управление конфигурацией включает следующие задачи:

1. Идентификация конфигурации (*Configuration Identification*);
2. Контроль конфигурации (*Configuration Control*);
3. Учет статуса конфигурации (*Configuration Status Accounting*);
4. Аудит конфигурации (*Configuration Audit*).

Управление конфигурацией для больших систем создается с помощью методов и средств, обеспечивающих идентификацию элементов этой системы, контроль вносимых изменений и возможность определения фактического состояния системы при разработке и эксплуатации в любой момент времени. Управление конфигурацией базируется на точной и достоверной информации о состоянии системы и планах проведения изменений. Для каждого возможного риска определяется показатель степени его вероятности и потерь, связанных с риском. Во время проведения *регрессионного тестирования* отыскиваются критические ошибки.

С формальной точки зрения управление конфигурацией состоит в дисциплинированном применении технического, административного управления и методов наблюдения за определенными и документированными функциональными и физическими характеристиками отдельных пунктов конфигурации и элементов системы, а также методов управления изменениями, подготовки отчетов по выполненным изменениям и процедурам их проверки на соответствие поставленным требованиям.

Работы по управлению конфигурацией, как правило, выполняет специальная служба, которая определяет возможные ограничения на функционирование системы в заданных условиях операционной среды, планирование внесения изменений, проверку разных частей системы, сбор данных и учет внесенных изменений в систему и конфигурацию. К деятельности этой службы относится также управление проектом, контроль качества и целостности конфигурации системы и ее сопровождение.

Структура службы зависит от сложности системы, этапов развития проекта и от специалистов организационноразработчика системы и заказчика. От хорошей организации работы службы зависит эффективность управления конфигурацией. Взаимосвязь видов деятельности по управлению конфигурацией дана [рис. 11.9](#).

Результатом управления конфигурацией является отчет о проведенных изменениях версии системы и документации, а также документ о передаче измененной версии пользователю.

Для достижения целей управления конфигурацией (УК) должно проводится планирование и выполнение проекта с учетом возникающих ограничений ОС и оборудования у заказчика. Процессом планирования занимаются менеджеры службы управления проектом. Предложения на изменение компонентов системы подаются в эту службу для проведения анализа и определения целесообразности внесения изменений в версию системы и ее конфигурацию.



Рис. 11.9. Виды деятельности управления конфигурацией

Процесс изменений включает в себя определение типов изменений, организацию их проведения и формирование концепции допуска отклонений и отказов по отношению к требованиям проекта системы.

Результат внесения изменений – новая версия *системы*, *документация* по проведению испытаний и пользовательская документация на систему. Заказчик оценивает предложения на внесение изменений и дает разрешение на проведение наиболее важных изменений, влияющих на технические характеристики или стоимость. Анализ и контроль проведения изменений конфигурации системы на этапах ЖЦ проводит специальная группа службы управления.

План изменений в конфигурацию системы утверждается формальными процедурами, расчетами оценок влияния изменений на стоимость, принятие решений об изменениях или отказ от них. Запросы на внесение изменений выполняются в соответствии с процедурами разработки системы на этапах ЖЦ или на этапе сопровождения системы. Поскольку требуемые изменения могут проводиться одновременно с разработкой,

предусматривается трассирование изменений при построении новых версий. Каждое проведенное изменение подвергается детальному аудиту.

При внесении изменений проводится контроль текущей версии системы с использованием репозитория, проверка исходного кода иполученной версий. Инструментами контроля являются Rational's ClearCase и SourceSafe of Microsoft системы Unix.

После завершения изменений и испытания системы проводится тиражирование системы и документации для передачи системы и ее конфигурации заказчику. В конфигурацию системы входят сведения об аппаратных и программных элементах системы. При этом на систему могут задаваться ограничения с учетом замечаний заказчика, аудиторов, а также информации из источников (спецификации требований, описания, отчеты и др.) и состава инструментальных средств и рекомендаций государственных или межведомственных стандартов.

Планирование конфигурации. Данный процесс зависит от типа проекта, организационных мероприятий, ограничений и общих рекомендаций по руководству конфигурацией. К видам планирования управления конфигурацией системы относятся: идентификация, определение статуса и аудита конфигурации, управление изменениями конфигурации.

При планировании составляются планы, выбираются инструменты, анализируются требования проекта, интерфейсы компонентов и т.п. К средствам обеспечения планирования относятся:

- система управления кодами, перевод и объединение компонентов в конфигурацию системы;
- базовые библиотеки и ресурсы;
- специальные группы контроля системы и ее конфигурации;
- СУБД для ведения проекта и хранения изменений.

К основным задачам планирования конфигурации относятся:

- фиксация разных заданий на изменения и выбор инструментария для их выполнения;
- определение человекочасов и инструментальных ресурсов, стандартов, затрат на внесение изменений и др.;
- установление связей с заказчиком для проведения контроля системы и конфигурации, а также проведение оценки системы;
- определение последовательности работ управления конфигурацией.

Результаты планирования отмечаются в плане управления конфигурацией проекта, а также в документе внесения изменений в версию, конфигурацию или в систему.

11.3.2. Идентификация элементов конфигурации

Идентификация конфигурации – это именованние всех элементов системы на основе схемы классификации и кодирования элементов, а также методов представления и ведения версий конфигурации с использованием входящих в нее элементов.

Она выполняется с помощью методов структуризации, классификации и именования элементов системы и ее версий. При этом проводится:

- определение стратегии идентификации для получения учтенной версии системы;
- именование составных элементов частей и всей конфигурации системы;
- установление соотношения между количеством выполняемых задач и количеством пунктов конфигурации;
- ведение версии системы (или ее частей) и документирование;
- выбор элементов базиса конфигурации и его формальное обозначение.

При идентификации используется библиотека элементов, версий и изменений системы. Основу идентификации составляет конфигурационный базис – набор формально рассмотренной и утвержденной конфигурационной документации как основы для дальнейшего развития или разработки системы.

Выделение в продукте контролируемых единиц конфигурации является составной частью процесса высокоуровневого или архитектурного проектирования и выполняется системными архитекторами. Построение адекватной схемы классификации и идентификации объектов *конфигурационного управления* выполняется одновременно со структуризацией продукта и заключается в определении:

- конфигурации продукта и ее версий;
- контролируемых единиц конфигурации и их версий;
- всех составляющих конфигурационного базиса и их редакций.

Результатом создания и применения схемы идентификации является возможность быстро и гарантированно различать разные продукты друг от друга, версии одного продукта между собой, единицы конфигурации продукта и их версии.

11.3.3. Управление версиями и контроль конфигурации

Версия системы включает элементы конфигурации и систему для передачи получателю [11.6, 11.7].

Управление версиями состоит в:

интеграции или композиции корректной и окончательной версии системы из элементов конфигурации, реализованных на этапах ЖЦ, а также с учетом аппаратных средств и инструментов построения системы;

выборе инструментов построения версии, оценке возможностей среды и средств автоматизации процесса построения отдельных версий с корректной конфигурацией ПО и данных;

управление вариантами версий, включающими совокупность готовых идентифицированных элементов системы и удовлетворяющих заданным требованиям заказчика к варианту.

При формировании версий системы учитываются ограничения на функции системы и отклонения от требований на разработку элементов конфигурации системы. После получения новой версии системы заказчику передаются: версия, конфигурация, документация и инструменты управления версиями для самостоятельного внесения изменений при сопровождении системы.

Примером формирования 21 версий ОС-360 (1965–1980 гг.) является фирма IBM. В каждую версию ОС постоянно добавлялись новые функциональные возможности путем внесения изменений в предыдущую версию после анализа ее экспериментальной эксплуатации [11.13]. Над развитием дополнительных возможностей этой ОС и внесением изменений в предыдущую версию постоянно работал коллектив фирмы. Трудоемкость разработки очередной версии ОС считалась пропорциональной интервалу времени между регистрируемыми очередными версиями и принималась за единицу измерения сложности создания новой версии [11.7].

В качестве меры трудоемкости сопровождения и создания очередной версии использовалось число модулей (ограниченных размеров со стандартизованным описанием), подвергающихся изменениям и дополнениям. Кроме того, оценивалась интенсивность работ по созданию версии, которая измерялась числом измененных модулей в единицу времени. После 12 лет постоянных изменений в ОС, 21-я версия работала стабильно, в нее почти не вносились изменения, так как отсутствовали претензии со стороны пользователей ОС. Метрический анализ процесса развития ОС-360 позволил установить, что объем среднего прироста системы на каждую версию соответствовал примерно 200 модулям. При этом общий объем увеличился от 1 тыс. модулей в первых версиях до 5 тыс. модулей в последних версиях. Когда уровень прироста сложности был большим, для устранения ошибок или дополнительных корректировок иногда создавались промежуточные версии с меньшим числом изменений.

В результате разработки ОС-360 было сформулировано понятие "критической массы" или критической сложности модифицируемой системы. Если при модернизации и выпуске очередной версии системы объем доработок превышает "критический", то возрастает вероятность ухудшения характеристик системы или необходимость введения промежуточной версии с внесением незначительных изменений. "Критический" объем доработок ОС-360 около 200 модулей оставался постоянным, несмотря на рост квалификации коллектива, совершенствование технических и программных средств и т.п. В первых версиях объем доработок составлял 20% модулей, а в последних версиях снизился до 5%.

Контроль конфигурации – это проверка и управление изменениями системы при формировании версии и эксплуатации. Процесс создания продукта включает непрерывные корректировки, которые имеют отношение к уже согласованному и/или утвержденному базису конфигурации. В этом плане предметом проведения контроля конфигурации являются:

изменения в базис конфигурации и связанная с ними корректировка конфигурации;

дефекты и отклонения в конфигурации продукта относительно утвержденного базиса.

Осуществляется это с помощью формальных процедур инициализации, анализа, принятия и контроля исполнения управленческих решений по изменениям, обнаруженным дефектам и отклонениям в конфигурации.

Формальная обработка запросов на изменение базиса. После того, когда заинтересованные участники проекта достигли взаимопонимания требований, архитектуры и других технических решений, соответствующие проектные документы считаются утвержденными и не могут произвольно модифицироваться. Т.е. любая потребность в изменении, исходящая от любого участника проекта, должна пройти такие формальные процедуры:

1. Регистрация предложения /запроса на изменение.
2. Анализ влияния предложенного изменения на имеющийся задел, объем, трудоемкость, график и стоимость работ по проекту.
3. Принятие решения на изменение (удовлетворить, отказать или отложить).
4. Реализация утвержденного изменения и его верификация.

Управление дефектами и отклонениями от утвержденного базиса. Второй составляющей контроля конфигурации – управление несоответствиями между конфигурацией или элементами продукта и базисом

конфигурации. С точки зрения управления все несоответствия принято делить на дефекты и отклонения. К дефектам относят те несоответствия, которые имеют непосредственное отношение к целевому использованию продукта по его назначению. Все остальное относится к отклонениям. Если дефекты в продукте носят негативный характер, то они подлежат устранению.

Для устранения дефектов и выявленных отклонений проводится:

- регистрация информации о полученном дефекте/отклонении;
- анализ и диагностика места и причины дефекта/отклонения, оценка объема, трудоемкости, сроков и стоимости переделок;
- принятие решения по устранению дефекта/отклонения, реализация и верификация этих недостатков.

Подобного рода решения – управленческие, их выполняют руководители или их полномочные представители. Как правило, уровень принятия решения по изменению программного продукта принимается путем согласования или утверждения документов для соответствующего базиса конфигурации.

Наиболее удобной формой реализации такого управленческого решения – руководящий совет по контролю конфигурации (*Configuration Control Board*).

11.3.4. Учет статуса и аудит конфигурации

Суть этого учета состоит в регистрации и предоставлении информации для эффективного управления конфигурацией. Предмет такого учета – информация о текущем статусе идентифицированных объектов *конфигурационного управления*, предложенных изменениях, а также о выявленных дефектах и отклонениях.

Отчет относительно статуса конфигурации – ключевой фактор для принятия управленческих решений по проекту. Оперативно регистрируемые и регулярно обновляемые данные учета статуса конфигурации – это исходные данные для формирования количественных оценок или метрик производительности и качества работ на проекте.

В системе учета статуса конфигурации накапливаются сводные отчеты о количестве обнаруженных и исправленных дефектов, поступивших и реализованных запросов на изменения, динамике внесения изменений в конфигурацию продукта и др. Такой отчетностью практически пользуются все участники проекта: заказчики, аналитики, разработчики, тестировщики, внедренцы, служба качества и руководство проекта.

Аудит конфигурации. Аудит – это ревизия или проверка очередной версии ПО перед сдачей системы заказчику, а также рассмотрение и оценка документации (данных, сводок, отчетов и др.). Он включает:

- функциональный аудит конфигурации, который проводится для подтверждения соответствия фактических характеристик конфигурации продукта требованиям заказчика;
- физический аудит конфигурации, являющийся подтверждением взаимного соответствия документации и фактической конфигурации продукта.

Функциональный аудит – это не верификация/валидация продукта, а проверка того, что тестирование проведено в установленном объеме, результаты документированы и подтверждают характеристики продукта, как соответствующие требованиям. При этом считается, что все изменения реализованы, дефекты устранены, а по отклонениям приняты адекватные решения. Независимые эксперты проводят аудит путем сверки выпущенного продукта с документами конфигурационного базиса, а также проверки правильности построения конфигурации в соответствии с заданными процедурами.

Контрольные вопросы и задания

1. Какие задачи решаются в менеджменте проекта?
2. Определите процесс планирования менеджмента проекта.
3. Определите понятие управления риском.
4. Объясните стратегию оценки стоимости продукта по Бозму.
5. Что понимается под процессом управления конфигурацией ПО?
6. Приведите основные задачи управления конфигурацией. Дайте общую характеристику понятий идентификации, учета статуса.
7. Какие действия выполняются в процессе управления версиями ПО?
8. Сформулируйте основные задачи учета и аудита конфигурации проекта.

Внимание! Если Вы увидите ошибку на нашем сайте, выделите её и нажмите Ctrl+Enter.