

Поток работ «Требования» (продолжение 3)

Артефакт: Субъект

Краткий обзор

Субъект определяет связанный набор ролей, которые пользователи могут играть при взаимодействии с системой. Пользователем может быть человек или внешняя система.

На иллюстрации на стр. 12-1 артефакт Субъект находится в зоне ответственности работника Проектировщик интерфейса пользователя. Действительно:

- **проектировщик интерфейса пользователя** отвечает за целостность описания субъектов-людей, гарантируя что:
 - Для каждого субъекта зафиксированы характеристики, необходимые для формирования интерфейса пользователя
 - Каждый субъект имеет правильные связи-ассоциации с прецедентами, с которыми он взаимодействует
 - Каждый субъект – часть правильных отношений обобщения
 - Каждый субъект определяет связующую роль и независим от других субъектов
 - Локальная диаграмма прецедентов, описывающая субъекта, читаема и совместима с другими свойствами
- **системный аналитик** имеет подобные обязанности для не-человеческих субъектов, кроме описания характеристик (первый пункт в этом списке).

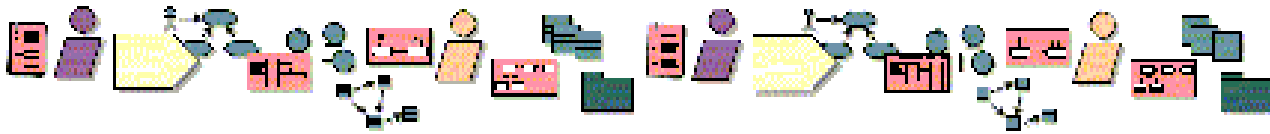
И все же впервые субъекты появляются из-под пера системного аналитика при выполнении им действия Поиск субъектов и прецедентов (см. иллюстрацию на стр. 11-7). Поэтому я думаю, что общие свойства этого артефакта нужно рассматривать в группе с артефактами, создаваемыми системным аналитиком.

Толкование

Экземпляр субъекта – это кто-то или что-то вне системы, что взаимодействует с системой.

Класс субъекта определяет набор экземпляров субъектов, в которых все экземпляры субъектов играют одинаковую роль относительно системы.

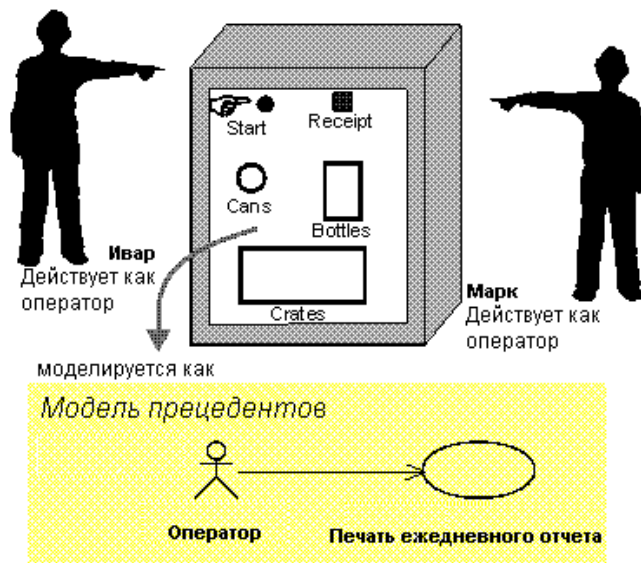
Чтобы полностью понять цель системы, Вы должны знать **для кого** предназначена эта система, то есть, кто будет использовать систему. Различные типы пользователей представляются как



субъекты.

Субъект – это что-то, что обменивается данными с системой. Субъектом может быть пользователь, внешние аппаратные средства или другая система.

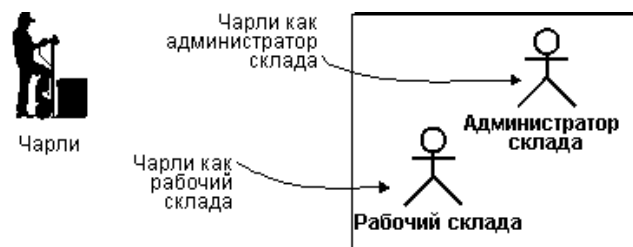
Разница между субъектом и индивидуальным пользователем системы состоит в том, что субъект представляет специфический класс пользователей, а не фактического пользователя. Несколько пользователей могут играть одну и ту же роль, что означает, что они могут быть одним и тем же субъектом. В этом случае каждый пользователь представляет экземпляр субъекта.



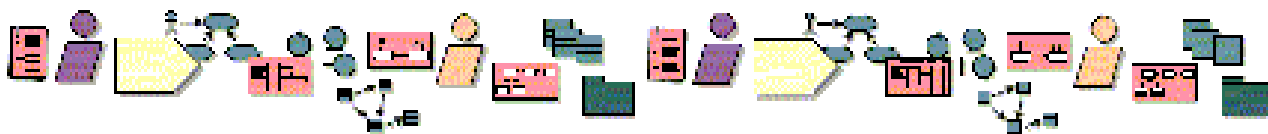
Ивар и Марк - операторы рециркуляторной машины. Когда они используют машину, каждый представлен экземпляром субъекта Оператор.

Однако, в некоторых случаях роль, смоделированную субъектом, играет только один человек. Например, для довольно маленькой системы может быть только один человек, играющий роль администратора системы.

Кроме того, один и тот же пользователь может действовать как несколько субъектов (то есть один и тот же человек может играть различные роли).



Чарли использует Систему складской обработки прежде всего как Администратора склада, но иногда он использует эту систему как обычный Работник склада.



Синхронизация

Субъекты отыскиваются и связываются с прецедентами в стадии Начало при определении границ системы. Характеристики субъектов описываются до того, как моделируются и реализуются интерфейсы пользователя.

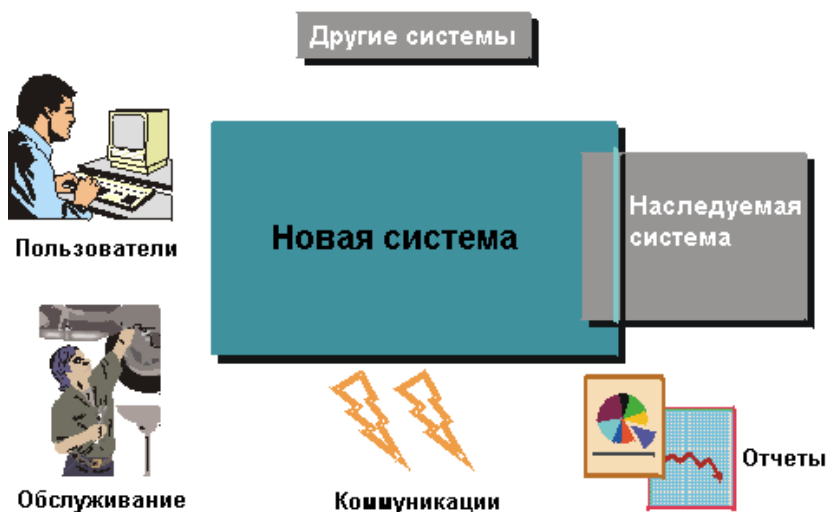
Создание и уточнение артефактов Субъект происходит в результате следующих действий Rational Unified Process:

- Поиск субъектов и прецедентов
- Моделирование интерфейса пользователя

Артефакт предоставляет информацию для следующих действий:

- Моделирование интерфейса пользователя
- Прототипирование интерфейса пользователя

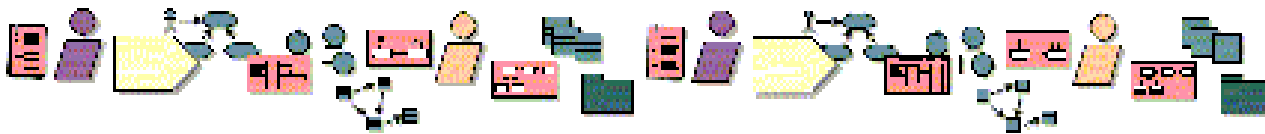
Как искать субъекты



Что из окружения системы станет субъектами системы?

Сначала нужно подумать о личностях, которые будут использовать систему. Как Вы можете их классифицировать? Часто бывает полезно представить в памяти несколько личностей (две или три) и удостовериться, что субъекты, которых Вы идентифицировали, покрывают их потребности. Когда Вы идентифицируете субъекты, полезно иметь в виду следующий набор вопросов:

- Кто поставляет, использует или удаляет информацию?
- Кто будет использовать эти функциональные возможности?



- Кто заинтересован в каких-либо требованиях?
- Где в организации используется система?
- Кто поддерживает и обслуживает систему?
- Каковы внешние ресурсы системы?
- Какие другие системы должны будут взаимодействовать с этой?

Имеется несколько различных аспектов окружения системы, которые Вы представите как отдельные субъекты:

- Пользователи, которые используют главные функции системы.

Пример:

У Системы складской обработки, которая поддерживает работу склада, есть несколько категорий пользователей: Рабочий склада, Служащий учета, Администратор склада. Все эти категории имеют в системе определенные роли, и поэтому Вы должны представить каждую из них отдельным субъектом.

- Пользователи, которые выполняют вторичные функции системы, типа системного администрирования.

Пример:

В рециркуляторной машине, используемой для приема банок, баллонов и контейнеров, главный субъект – это Заказчик, для которого прежде всего и сформирована система. Однако, кто-то должен управлять машиной. Эта роль отводится субъекту – Оператору.

- Внешние аппаратные средства, используемые системой.

Пример:

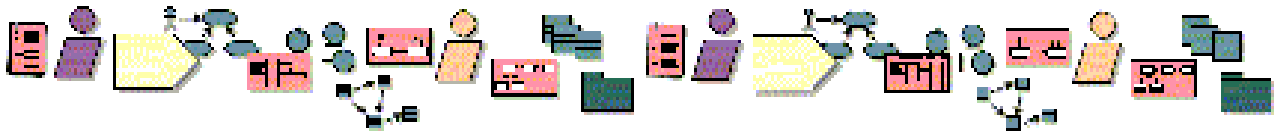
Система вентиляции, которая управляет температурой в помещении, непрерывно контролирует данные датчиков температуры. Поэтому Датчик – это субъект.

- Другие системы, взаимодействующие с системой.

Пример:

Банкомат должен связываться с центральной системой, которая проводит банковские счета. Центральная система – вероятно, внешняя для банкомата, и поэтому должна быть субъектом.

Если Вы строите прикладную программу для интернета, ваши основные субъекты будут в некотором смысле анонимными. Вы действительно не знаете, кем они являются, и Вы не можете сделать никаких предположений относительно их навыков и квалификации. Но Вы все-таки можете описать роль, которую, как Вы ожидаете, они будут играть в вашей системе.



Пример:

Системы, которые предоставляют информацию (типа механизмов поиска), будут иметь вполне анонимных субъектов, которые обращаются к приложению только для того, чтобы найти информацию относительно конкретной темы.

Пример:

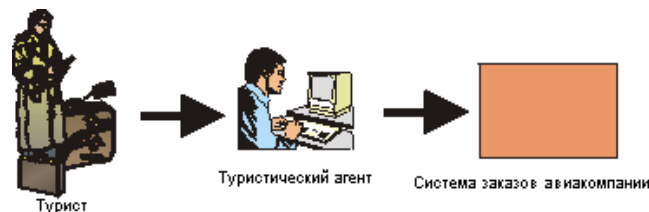
Правительственный информационный сайт, чей регламент состоит в предоставлении любому гражданину или абоненту сети информации относительно законов и правил, их применения, форм и так далее. Например, Налоговое управление США имеет страницу, которая предоставляет информацию о правилах заполнения налоговой декларации. Она предоставляет все формы, доступные в электронном виде, а также позволяет абонентам зарегистрировать свою налоговую декларацию с помощью интернета. Роль основного субъекта в этом случае играет любой, заинтересовавшийся тем, как регистрируется налоговая декларация в США. Конечно, человек, однажды зарегистрировавший свое посещение этой страницы, больше не является анонимным.

Субъекты помогают определять границы системы

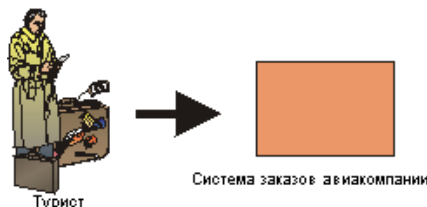
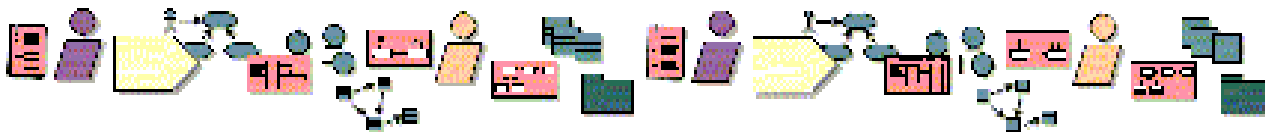
Обнаружение субъектов также означает, что Вы установили границу, которая помогает в понимании цели и контекста системы. Как субъекты должны рассматриваться только те, кто непосредственно связывает свои потребности с системой. Если Вы включаете большее количество ролей, чем их существует в окружении системы, Вы будете моделировать деловую сферу, в которой система будет использоваться опосредованно.

Пример:

Кто является субъектом в Системе заказов авиакомпании? Это зависит от того, строите ли Вы систему, которую будет использовать туристический агент, или Вы строите систему, с которой пассажир может соединяться непосредственно через интернет.



Если Вы строите Систему заказов, которая будет использоваться в туристическом агентстве, субъектом будет Туристический агент. Турист не взаимодействует непосредственно с системой, и поэтому он не субъект



Если Вы строите систему, которая позволит пользователям соединяться с ней через интернет, Турист взаимодействует непосредственно с системой, и поэтому он является субъектом системы.

Отчет

Отчет об артефакте Субъект используют:

- **системные аналитики** при фиксации границ системы;
- **авторы прецедентов** при описании прецедентов и их взаимодействия с субъектами;
- **проектировщики интерфейса пользователя** при фиксации характеристик субъектов – людей
- **аналитики объектов** при реализации прецедентов и их взаимодействия с субъектами.

Интерактивная версия Rational Unified Process содержит предопределенный шаблон и инструкции по созданию отчета о субъекте с использованием SoDA. SoDA автоматизирует генерацию отчета на основе информации модели прецедентов Rational Rose так, чтобы он был создан быстро и точно. Отчет о субъекте может быть сгенерирован SoDA с версиями программ Word или Frame.

Отчет состоит из следующих разделов

1. Краткое описание

Краткое описание субъекта должно включать следующую информацию:

- Кого или что представляет субъект.
- Почему необходим субъект.
- Какие интересы субъект имеет в системе.

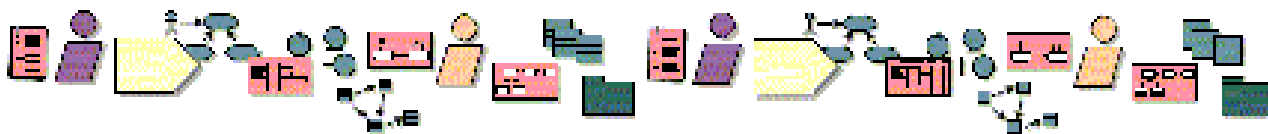
Краткое описание должно состоять, самое большее, из нескольких предложений.

Пример:

В модели прецедентов рециркуляторной машины три субъекта кратко описаны следующим образом:

Заказчик: Заказчик собирает дома баллоны, банки и контейнеры и возвращает их в магазин, чтобы получить деньги обратно.

Оператор: Оператор отвечает за эксплуатацию рециркуляторной машины.



Администратор: Администратор отвечает за вопросы обеспечения деньгами и за обслуживание заказчика в магазине.

2. Характеристики

Характеристики субъекта могут влиять на разрабатываемую систему и, в особенности, на оптимально подходящую форму интерфейса пользователя. Обратите внимание, что, если деловые работники, соответствующие субъектам, уже описаны в модели деловых объектов, некоторые из следующих характеристик, возможно, уже были зафиксированы. Характеристики субъекта включают:

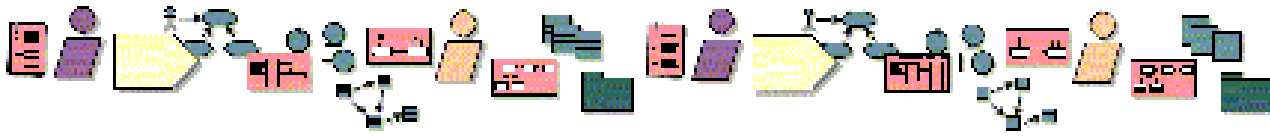
- Область ответственности субъекта.
- Физическая среда, в которой субъект будет использовать систему. Отклонения от идеального случая (когда пользователь находится в тихом офисе, не отвлекается), которые могут затрагивать использование таких вещей как звук, выбор шрифта и надлежащего применения комбинаций устройства ввода данных (например, клавиатуры, сенсорного экрана, мыши и быстрых клавиш).
- Число пользователей, представленных этим субъектом. Это число – полезный фактор при определении значения субъекта и значения частей интерфейса пользователя, который субъект использует.
- Частота, с которой субъект будет использовать систему. Эта частота определит, как долго (интерфейсу пользователя) нужно ожидать субъекта и сохранять его информацию между сеансами.

В большинстве случаев достаточно приблизительная оценка числа пользователей и частоты использования. Разница между 30 и 40 не будет затрагивать форму интерфейса пользователя, но разница между 3 и 30 уже может влиять на нее.

Другие характеристики субъекта включают:

- Уровень знаний субъектом проблемной области. Этот уровень поможет определять необходимые размеры проблемной справки, допустимость использования в интерфейсе пользователя проблемной терминологии.
- Уровень компьютерного опыта субъекта. Этот уровень поможет правильно определить соотношение сложных и упрощенных методов взаимодействия в интерфейсе пользователя.
- Другие прикладные программы, используемые субъектом. Заимствование концепций интерфейса пользователя от этих приложений сократит время обучения субъекта и уменьшит нагрузку на его память, так как субъект уже знаком с этими концепциями.
- Общие характеристики субъектов, типа уровня знаний (образования), знания языков, возраста и т.д. Эти характеристики могут влиять на подробности интерфейса пользователя, тип шрифта и языка.

Эти характеристики используются, прежде всего, при идентификации граничных классов и



прототипа, чтобы гарантировать лучшее соответствие применимости между семейством пользователя и проектом интерфейса пользователя.

Пример:

Ниже приводится пример характеристик субъекта Пользователь почты. Этот субъект, среди прочего, взаимодействует с прецедентом Управление входящим сообщением почты.

- Пользователь почты - опытный пользователь ПК.
- Рабочая среда Пользователя почты - обычно тихий офис.
- Целевое число пользователей почты – ок. 500,000.

3. Связи

Перечисляются связи, включающие субъекта.

- Для связи-ассоциации включаются ее краткое описание и связанные прецеденты. Включаются также их кратность и навигация из роли, включающей рассматриваемый субъект.
- Для обобщения включается его краткое описание и связанный субъект-предок.

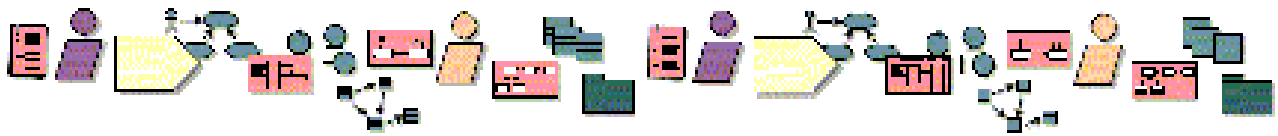
4. Диаграммы

Диаграммы субъекта и его связей.

Контрольные точки

Качество артефактов Субъект необходимо оценить на их соответствие следующим контрольным точкам:

- Все ли субъекты найдены? То есть, все ли роли в среде системы Вы объяснили и смоделировали? Хотя Вы и должны проверить это, Вы не можете дать окончательного ответа, пока Вы не нашли и не описали все прецеденты.
- Каждый ли субъект связан, по крайней мере, с одним прецедентом? Удалите все субъекты, не упомянутые в описаниях прецедентов, не имеющие связей-ассоциаций с прецедентами. Однако, субъект, упомянутый в описании прецедента, вероятно, будет иметь связь-ассоциацию с этим прецедентом.
- Можете ли Вы назвать, по крайней мере, двух людей, которые могли бы играть роль конкретного субъекта? Если нет, проверьте, не является ли роль субъекта частью другой роли. Если да, то объедините субъект с другим субъектом.
- Имеются ли субъекты, которые играют относительно системы подобные роли? Если да, Вы должны объединить их в один субъект. Связи-ассоциации и описания прецедента показывают, как взаимодействуют субъекты и система.
- Имеются ли два субъекта, играющие относительно прецедента одну и ту же роль? Если да, Вы должны использовать обобщение субъектов, чтобы моделировать их одинаковое



поведение.

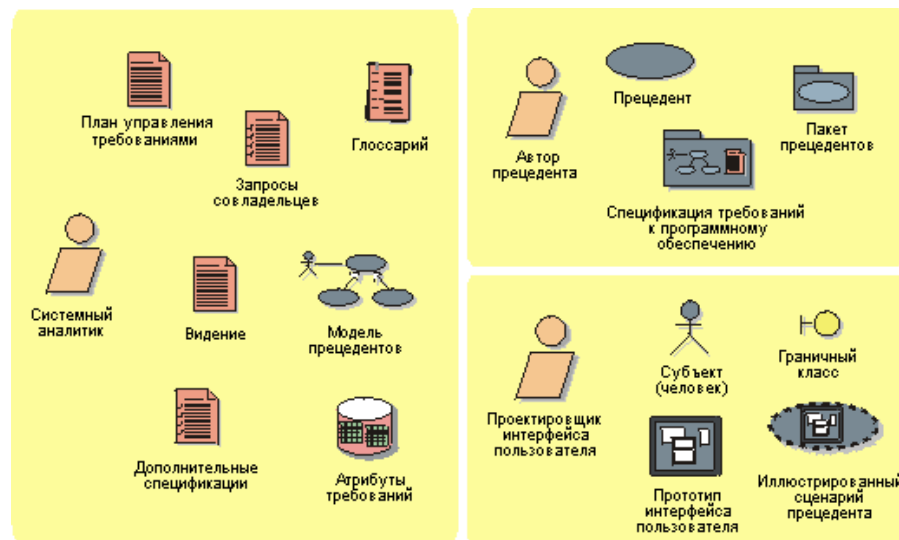
- Будет ли конкретный субъект использовать систему несколькими (полностью различными) способами, или он имеет несколько (полностью различных) целей в использовании прецедента? Если да, то Вы, вероятно, имеете больше одного субъекта.
- Имеют ли субъекты интуитивно понятные и подробные названия? Пользователи и заказчики понимают эти названия? Важно, чтобы название субъекта соответствовало его роли. Если нет, измените название.

Артефакт: План управления требованиями

Краткий обзор

Артефакт План управления требованиями описывает документацию требований, типы требований и соответствующие им атрибуты требований, определяет информацию и механизмы управления, которые будут использоваться для измерения, составления отчетов и управления изменением требований к изделию.

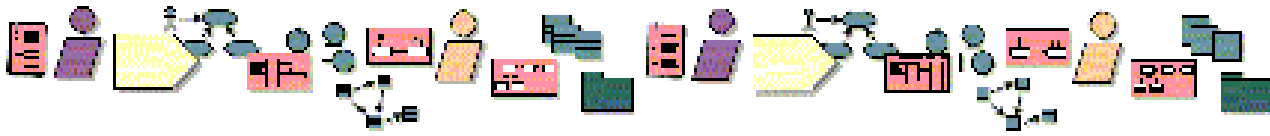
Этот очень важный артефакт, с которого, собственно, должна начинаться работа по управлению требованиями, является вложением в составной артефакт План разработки программного обеспечения (см. иллюстрацию на стр. 6-11). В предыдущих версиях План управления требованиями не выделялся в самостоятельный артефакт. Теперь эта несправедливость устранена:



Артефакты потока работ «Требования» Rational Unified Process 2000. Сравните с иллюстрацией на стр. 12-1.

Синхронизация

Артефакт План управления требованиями разрабатывается в стадии Начало и модифицируется в каждой главной вехе.



Создание и модификации артефакта происходят в результате следующих действий Rational Unified Process:

- Разработка плана управления требованиями
- Управление зависимостями

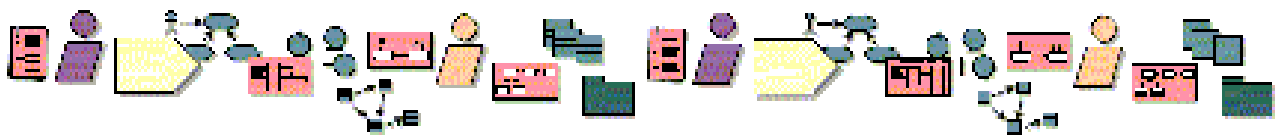
Артефакт предоставляет информацию для следующих действий:

- Составление плана разработки программного обеспечения
- Детализация прецедента
- Детализация требований к программному обеспечению
- Внедрение адаптированного процесса
- Управление зависимостями

Отчет

Интерактивная версия Rational Unified Process содержит predetermined шаблон и инструкции по его использованию для артефакта План управления требованиями. Ниже приводится его краткая схема:

1. **Введение**
 - 1.1 **Цель**
 - 1.2 **Контекст**
 - 1.3 **Определения, акронимы и сокращения**
 - 1.4 **Ссылки**
 - 1.5 **Краткий обзор**
2. **Артефакты требований и типы требований**
3. **Атрибуты требований**
 - 3.1 **Атрибуты для <тип требования>**
 - 3.1.1 **Статус**
 - 3.1.2 **Выгода**
 - 3.1.3 **Объем работ**
 - 3.1.4 **Риск**
 - 3.1.5 **Стабильность**



3.1.6 Целевой выпуск

3.1.7 Кому назначен

3.1.8 Причина

4. Критерии трассируемости

4.1 Критерий для <тип требования>

Работник: Автор прецедента

Человек, действующий как автор прецедента, должен иметь хорошие навыки общения с людьми и навыки записи результатов бесед. Знание прикладной области желательно, но не необходимо для всех. Автор прецедента должен быть знаком со всеми инструментальными средствами и иметь опыт фиксации результатов потока работ Требования

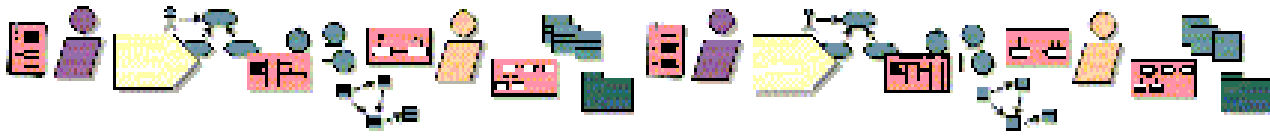
Артефакт: Прецедент

Краткий обзор

Прецедент определяет набор экземпляров прецедента, где каждый экземпляр – это последовательность действий, выполняемых системой, которая выдает наблюдаемый результат, ценный для конкретного субъекта.

Прецеденты используют следующие категории совладельцев и участников работ:

- **Заказчики** используют прецеденты, чтобы понять поведение системы, и, так как они должны согласовывать поток событий прецедента, заказчики используют прецеденты при рассмотрении результатов моделирования.
- **Потенциальные пользователи** используют прецеденты, чтобы понять поведение системы.
- **Архитекторы** используют прецеденты, чтобы идентифицировать архитектурно существенные функциональные возможности.
- Те, кто **анализирует, проектирует и выполняет** систему, используют прецеденты, чтобы понять требуемое поведение системы и уточнять систему.
- **Проектировщики прецедентов** используют потоки событий прецедентов для поиска классов. (Для проектировщиков прецедентов они являются наиболее важными артефактами.)
- **Тестировщики** используют прецеденты как базу для идентификации вариантов тестирования.
- **Администраторы** используют прецеденты для планирования и отслеживания моделирования прецедентов.
- **Авторы документов** используют прецеденты, чтобы понять, как описывать в документации



последовательность использования системы (типа руководства пользователя).

Толкование

Определение понятия прецедента и подробное его толкование уже приводилось в главе 4, при обсуждении основных концепций Rational Unified Process (см. стр. 4-5).

Синхронизация

Прецеденты идентифицируются и по возможности кратко описываются как можно раньше в стадии Начало, чтобы помочь в определении возможностей системы. Прецеденты, необходимые для выполнения анализа или архитектурного проектирования системы, более подробно описываются в стадии Уточнение. Оставшиеся прецеденты подробно описываются в стадии Конструирование.

Поиск, создание и модификации прецедентов происходят в результате следующих действий Rational Unified Process:

- Поиск субъектов и прецедентов
- Структурирование модели прецедентов
- Детализация прецедента
- Обзор требований

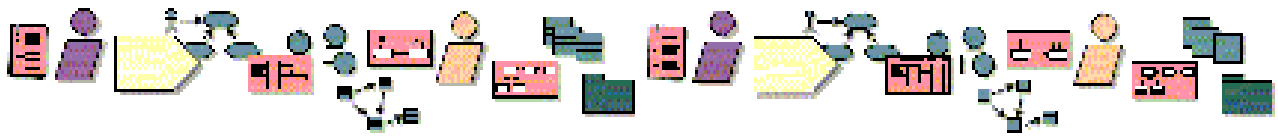
Прецедент предоставляет информацию для следующих действий:

- Фиксация общего словаря
- Детализация прецедента
- Детализация требований к программному обеспечению
- Моделирование интерфейса пользователя
- Обзор требований
- Структурирование модели прецедентов
- Анализ прецедента
- Проектирование прецедента

Как искать прецеденты

Ниже представлен набор вопросов, на которые полезно ответить при идентификации прецедентов:

- Для каждого ли субъекта определены все его задачи, для решения которых должна использоваться система?
- Нужно ли информировать субъекта относительно наступления некоторых событий в системе?



- Должен ли субъект сообщать системе о внезапных внешних изменениях?
- Обеспечивает ли система правильное поведение в деловой сфере?
- Все ли возможности могут быть выполнены идентифицированными Вами прецедентами?
- Какие прецеденты поддерживают и обслуживают систему?
- Какая информация должна изменяться или создаваться в системе?

Прецеденты, которые часто пропускаются, так как они не отражают первичные функции системы, могут быть следующего вида:

- Запуск и останов системы.
- Эксплуатация системы. Например, добавление новых пользователей и установка параметров пользователя.
- Эксплуатация данных, хранящихся в системе. Например, система создана для работы параллельно с наследуемой системой, и информация должна быть синхронизирована между ними.
- Функциональные возможности, необходимые для изменения поведения системы. Примером может служить функциональная возможность создания новых отчетов, а не только жесткого набора определенных отчетов о данных, хранящихся в системе.

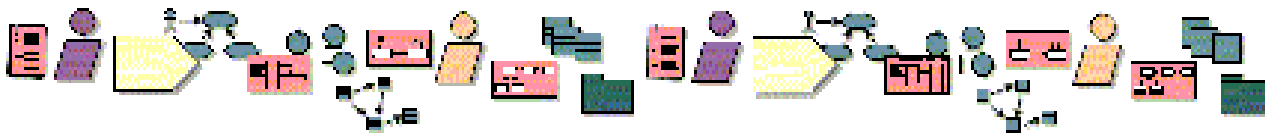
Если Вы разработали модели деловых прецедентов и деловых объектов, используйте имеющуюся в них информацию (см. стр. 10-17).

Как эволюционируют прецеденты

В ранних итерациях стадии Уточнение детально описывают только несколько прецедентов, которые рассматриваются как архитектурно существенные. Всегда сначала нужно разрабатывать схему прецедента (в пошаговом формате), прежде, чем углубляться в подробности. Эта пошаговая схема должна быть вашей первой попыткой определения структуры потока событий прецедента (см. раздел Поток событий - структура на стр. 14-17). Всегда начинайте с основного потока прецедента. Когда достигается некоторое соглашение по схеме основного потока, Вы можете добавлять то, что вводят альтернативные потоки относительно основного потока.

К концу уточнения все прецеденты, которые Вы планируете описать в деталях, должны быть закончены.

В вашей модели будут встречаться прецеденты, которые настолько просты, что они не нуждаются в детальном описании потока событий, вполне достаточно и пошаговой схемы. Критерием для принятия решения является то, что Вы не видите непонимания среди читателей, и что проектировщики и тестировщики согласны с уровнем детализации, предоставляемым пошаговым форматом. Примеры – прецеденты, которые описывают простой ввод или поиск некоторых данных в системе.



Контекст прецедента

Часто трудно решить, рассматривать ли некоторый набор взаимодействий системы и пользователя, как один или как несколько прецедентов. Рассмотрите использование рециркуляторной машины. Заказчик ставит пустую тару, типа банок, баллонов и контейнеров, в рециркуляторную машину. Когда он поставил всю имеющуюся тару, он нажимает кнопку, после чего печатается чек. Теперь он может обменять этот чек на деньги.

Как быть? Считать одним прецедентом установку тары и другим – затребование чека? Или все это – один прецедент? Происходит два действия, но одно без другого имеет для заказчика небольшое значение. Скорее всего, это – один законченный диалог с установкой тары и получением чека, который является ценным для заказчика (и имеет для него смысл). Таким образом, законченный диалог, от установки первого предмета до нажатия кнопки и получения чека, является законченным прецедентом.

Кроме того, Вы захотите сохранить эти два действия вместе, иметь возможность делать их одновременный обзор, вместе их изменять, тестировать, составлять для них справочники и вообще управлять ими как модулем. Это становится очевидным в больших системах.

Как реализуются прецеденты

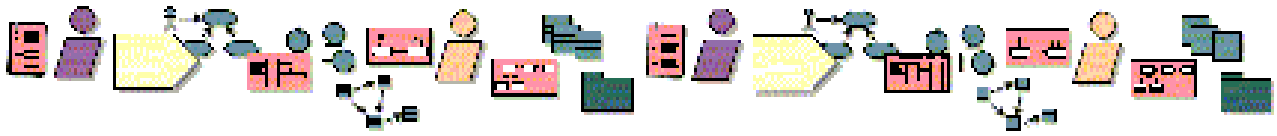
Прецедент описывает то, что происходит с системой, когда субъект взаимодействует с ней для выполнения прецедента. Прецедент не определяет то, как внутри системы выполняются ее задачи в терминах сотрудничества объектов. Это должна показывать реализации прецедента.

Пример:

В примере с телефоном, прецедент указал бы – среди прочего – что когда абонент поднимает трубку, система дает гудок, затем принимает набор цифр, находит сторону приема, там звонит телефон, при подъеме трубки происходит соединение, передается разговор и так далее.

В выполняемой системе образец прецедента не соответствует какому-либо конкретному объекту в модели выполнения (например, образцу класса в коде). Вместо этого, он соответствует определенному потоку событий, который вызван субъектом и выполнен как последовательность событий в наборе объектов. Другими словами, образцы прецедентов соответствуют поддерживающим связь образцам реализованных объектов. Мы называем это реализацией прецедента. Часто одни и те же объекты участвуют в реализации более, чем одного прецедента. Например, прецеденты Депонирование и Снятие со счета в банковской системе могут использовать в своих реализациях некоторый объект Счет. Но это не подразумевает, что два прецедента связаны, просто в своих реализациях они используют один и тот же объект.

Вы можете рассматривать поток событий как состоящий из нескольких подпотоков, которые вместе образуют суммарный поток событий. Вы можете многократно использовать описание подпотока в потоках событий других прецедентов. Подпотоки в описании потока событий одного прецедента могут быть общими с таковым других прецедентов. При проектировании Вы должны



иметь одни и те же объекты, выполняющие это общее поведение для всех соответствующих прецедентов; то есть это поведение должен выполнить только один набор объектов, независимо от того, который из прецедентов выполняется.

Пример:

В прецедентах Снятие денег и Проверка баланса системы банкомата начальный подпоток один и тот же. Поток событий обоих прецедентов начинается с проверки допустимости кредитной карты и персонального кода пользователя.

Экземпляр прецедента может следовать почти неограниченным, но счетным числом путей. Эти пути определяют выборы, обозначенные для экземпляра прецедента в описании потока событий. Выбранный путь зависит от событий. Типы событий включают:

- Ввод от субъекта. Например, субъект может выбрать из нескольких предложенных вариантов, что делать дальше.

Пример:

В прецеденте Возврат предметов в системе рециркуляторной машины Заказчик всегда имеет две возможности: его рука все еще ставит в машину очередной предмет или нажимает кнопку и получает чек.

- Проверка значения или типа внутреннего объекта или атрибута. Например, поток событий может отличаться, если значение больше или меньше, чем некоторое значение.

Пример:

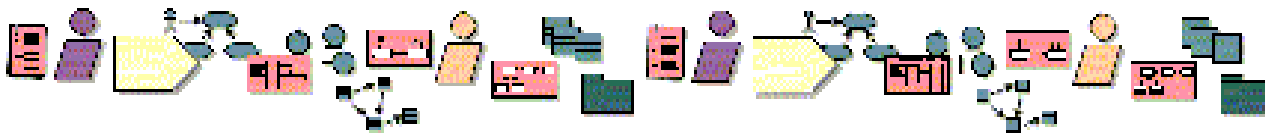
В прецеденте Снятие денег в системе банкомата поток событий будет иным, если Пользователь просит выдать денег больше, чем он имеет на своем счете. Таким образом, экземпляр прецедента будет следовать различными путями.

Параллельное выполнение экземпляров прецедента

Экземпляры нескольких прецедентов и несколько экземпляров одного и того же прецедента работают одновременно, если система это позволяет. При моделировании прецедента Вы можете предполагать, что экземпляры прецедентов могут без конфликта быть одновременно активными. Проектная модель, как ожидается, решит эту проблему, потому что модель прецедента не описывает, как система работает. Один из способов рассмотрения состоит в том, чтобы предполагать, что одновременно активен только один экземпляр прецедента и что выполнение этого экземпляра является элементарным действием. При моделировании прецедента предполагается, что машина интерпретации работает бесконечно быстро, так, чтобы преобразование экземпляров прецедента в последовательную цепочку не было проблемой.

Название и краткое описание

Каждый прецедент должен иметь название, которое указывает на то, что достигается его взаимодействием с субъектом (субъектами). Название, вероятно, будет состоять из нескольких слов, которые будут понятны. Никакие два прецедента не могут иметь одно и то же название.



Пример:

Здесь представлены варианты названия для прецедента Возврат предметов в примере с рециркуляторной машиной:

- Прием предметов залога
- Возврат предметов залога
- Предметы залога

Краткое описание прецедента должно отражать его роль и назначение. Когда Вы записываете описание, обратитесь к субъектам, включенным в прецедент, к глоссарию и, если необходимо, определите новые понятия.

Пример:

Ниже представлены примеры краткого описания прецедентов Возврат предметов и Добавление нового типа баллона в системе рециркуляторной машины:

Возврат предметов: Пользователь использует машину для автоматического подсчета всех сданных предметов залога (баллонов, банок и контейнеров) и получения чека. Чек должен быть оплачен в кассе.

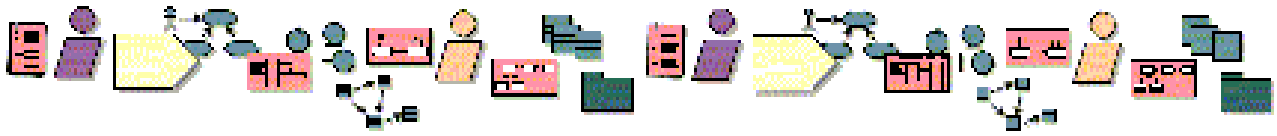
Добавление нового типа баллона: Новые типы баллонов могут быть добавлены к системе запуском «режима обучения» и вставкой пяти баллонов этого типа точно так же, как при обычном возвращении предметов. Машина может измерить баллоны и «научиться» их идентифицировать. Администратор определяет сумму возмещения для нового типа баллонов.

Поток событий – содержание

Поток событий прецедента содержит наиболее важную информацию, производимую при моделировании прецедента. Он должен описывать поток событий прецедента достаточно ясно и просто для постороннего. Помните, что поток событий должен представлять то, что делает система, а не то, как сконструирована система, чтобы выполнять требуемое поведение.

Рекомендации по содержанию потока событий:

- Опишите, как прецедент начинается и заканчивается
- Опишите, какими данными обмениваются субъект и прецедент
- Не описывайте подробности интерфейса пользователя, если это не необходимо для понимания поведения системы. Например, когда известно заранее, что приложение будет базироваться на сетевой технологии, полезно использовать ограниченный набор сетевой терминологии,. Иначе текст прецедента может показаться слишком абстрактным. При изложении используйте слова «переходит», «просматривает», «гиперсвязь», «страница», «подтверждает» и «окно просмотра». Однако, не желательно включать в описание слова «фрейм» или «web-страница», т.к. этим Вы высказываете предположения относительно проектного решения.

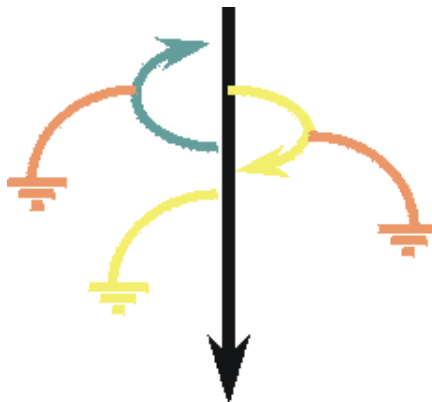


- Описывайте поток событий, а не только функциональные возможности. Чтобы выполнить это, начинайте описание каждого действия со слов «Когда субъект ...»
- Описывайте только те события, которые принадлежат прецеденту, а не то, что случается в других прецедентах или вне системы
- Избегайте неопределенной терминологии типа «например», «и т.д.» или «информация о ...»
- Детализируйте поток событий. Нужно ответить на все «когда» и «если». Помните о тех проектировщиках испытаний, которые будут использовать этот текст для идентификации вариантов тестирования.

Если Вы использовали некоторые термины в других прецедентах, убедитесь, что были использованы точно те же термины, и что их значение является точно тем же. Чтобы управлять общими терминами, поместите их в глоссарий.

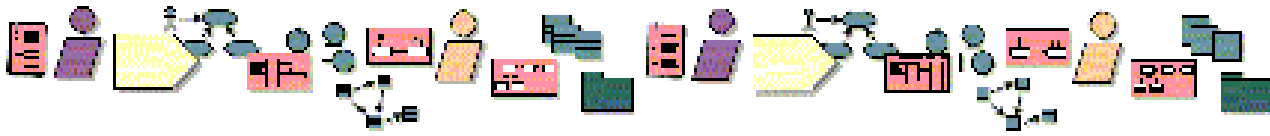
Поток событий – структура

Поток событий может содержать **основной поток событий** и **альтернативные потоки событий**. Основной поток событий должен отражать то, что «обычно» случается при выполнении прецедента. Альтернативные потоки событий отображают поведение необязательного или исключительного характера относительно нормального поведения, а также вариации нормального поведения. Вы можете рассматривать альтернативные потоки событий как «окольные пути» относительно основного потока событий, некоторые из которых возвратятся к основному потоку событий, а некоторые – закончат выполнение прецедента.



Типичная структура потока событий. Прямая стрелка представляет основной поток событий, а кривые – альтернативные пути относительно основного. Некоторые альтернативные пути возвращаются к основному потоку событий, в то время как другие заканчивают прецедент.

И основной поток событий, и альтернативные потоки должны быть далее структурированы в шаги или подпотоки. При этом ваша главная цель должна состоять в составлении понятного текста (см. также раздел Поток событий - стиль ниже). Практическое правило состоит в том, что подпоток должен быть сегментом поведения в пределах прецедента, который имеет ясную цель



и атомарность в том смысле, что Вы выполняете или все или ни одного из описанных действий. Вы можете иметь несколько уровней подпотоков, но если это возможно, лучше стараться избегать этого, так как усложняется понимание текста. Вы можете иллюстрировать структуру потока событий с помощью диаграммы действий (см. стр. 14-26).

Этот способ написания текста, структурированного в последовательные подразделы, рассчитан на читателя, который знает порядок следования подпотоков. Чтобы избежать недоразумений, Вы должны всегда указывать, установлен ли порядок подпотоков или нет. Соображения этого вида часто имеют связь с:

- Деловыми правилами. Например, права пользователя должны быть проверены прежде, чем система сделает некоторые данные доступными.
- Проектом пользовательского интерфейса. Например, система не должна приводить в исполнение некоторую последовательность поведения, которая может быть интуитивной для одних, но не для других пользователей.

Чтобы показать в структуре место альтернативного потока событий, Вы должны представить следующую информацию для каждого «окольного пути» относительно основного потока событий:

- Где в основном потоке событий может быть вставлено альтернативное поведение.
- Условие, которое должно быть выполнено, чтобы началось альтернативное поведение.
- Как и где продолжается основной поток событий, или как заканчивается прецедент.

Пример:

Здесь описан альтернативный подпоток прецедента Возврат предмета в системе рециркуляторной машины.

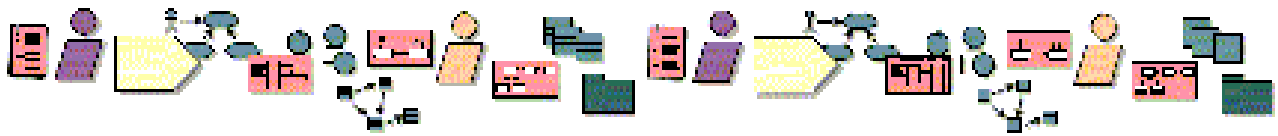
2.1. Баллон застрял

Если на шаге 1.5, Вставить возвращаемый предмет, баллон застревает в гнезде, датчики гнезда обнаружат эту проблему. Конвейерная лента останавливается и машина дает сигнал тревоги для вызова оператора. После устранения проблемы машина продолжает выполнение основного потока с шага 1.9.

В этом примере альтернативный поток событий вставлен в определенном месте основного потока событий. Но бывают такие альтернативные потоки событий, которые могут быть вставлены больше, чем в одном месте, а некоторые могут быть вставлены даже в любом месте основного потока событий.

Пример:

Ниже описан альтернативный подпоток прецедента Возврат предмета в системе рециркуляторной машины:



2.2. Передняя панель снята

Если кто-то снимет переднюю панель рециркуляторной машины, машина перестает работать и дает сигнал тревоги оператору. Когда передняя панель снова закрыта, машина продолжает работу с того места в основном потоке событий, где она была остановлена.

Если альтернативный поток событий достаточно прост, то может возникнуть соблазн описывать его в разделе основного потока событий, используя некоторую неофициальную конструкцию типа «условного оператора». Этого нужно избегать. Слишком много альтернатив сделают плохо различимым нормальное поведение. Кроме того, включение в разделы основного потока событий альтернативных путей может сделать текст «псевдокодоподобным» и трудно читаемым.

В общем случае, извлекая части потока событий и описывая эти части отдельно, можно увеличивать разборчивость основного потока событий и улучшать структуру прецедента и модели прецедентов. Вы можете моделировать извлеченные части:

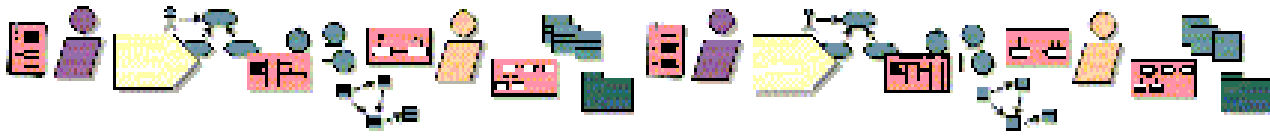
- Как альтернативный поток событий в пределах основного прецедента, если это – просто вариант, выбор или исключительная ситуация в основном потоке событий.
- Как явное включение в основной прецедент, если это – что-то, что Вы хотите сформировать так, чтобы его могли многократно использовать другие прецеденты.
- Как неявное включение в основной прецедент, если основной поток событий основного прецедента закончен, то есть имеет определенные начало и конец. Расширяющий поток делается тогда, когда Вы предпочитаете не показывать его в описании основного прецедента, чтобы не делать этот прецедент слишком сложным.
- Как подпоток в основном потоке событий, возможно как другой выбор, если не применяется ни одна из вышеупомянутых альтернатив. Например, в прецеденте Поддержка информации служащего могут быть отдельные подпоток для добавления, удаления и изменения информации о служащем.

Поток событий – стиль

Для описания прецедентов существует множество стилей. Для примера мы показываем основной поток событий прецедента Администрирование регламента, описанный тремя различными стилями, отличающимися, прежде всего, степенью формальности. Первый стиль, показанный в примере 1, рекомендуется потому, что он прост для понимания, и порядок, в котором случаются различные вещи, очевиден. Текст разделен на пронумерованные и имеющие названия подразделы. Нумерация упрощает ссылки на подразделы. Названия позволяют читателю быстро делать краткий обзор потока событий, просматривая текст и читая только заголовки.

Описание потока событий в примере 2 не в состоянии указать порядок, в котором случаются различные вещи. Если Вы запишете в этом стиле, Вы и другие можете упустить некоторые важные вещи, которые касаются системы.

Другой стиль, показанный в примере 3, может быть полезен, если Вам трудно выразить



последовательность событий. Этот стиль псевдокода более точен, но текст получается сложным для чтения и понимания человеком, не обладающим навыками программирования, особенно, если Вы хотите добиться быстрого понимания потока событий.

Пример 1:

1.1. Начало прецедента

Прецедент начинается, когда субъект Оператор требует от системы создать регламент измерения. Система ищет все элементы сети субъекта, их измеряемые объекты и соответствующие функции измерения, которые доступны этому конкретному Оператору. Доступные элементы сети – те, которые находятся в работе, и на обращение к которым Оператор имеет полномочия. Доступность функций измерения зависит от того, что было установлено для конкретного типа измеряемого объекта.

1.2. Конфигурирование регламента измерения

Система позволяет субъекту Оператор выбирать элементы сети для измерения, а затем показывает, какие измеряемые объекты доступны для выбранных элементов сети. Система позволяет Оператору выбрать измеряемые объекты, а затем выбрать, какие функции измерения устанавливать для каждого измеряемого объекта.

Система позволяет Оператору вводить текстовый комментарий относительно регламента измерения.

Оператор сообщает системе о завершении формирования регламента измерения. Система отвечает, генерируя уникальное название для регламента измерения и устанавливая значения по умолчанию для частоты и продолжительности измерения. Значения по умолчанию уникальны для каждого Оператора. Затем система позволяет Оператору редактировать эти значения по умолчанию.

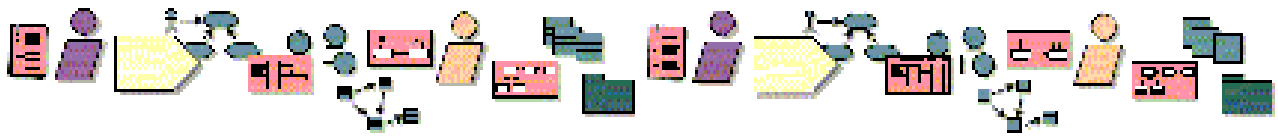
1.3. Инициализация регламента

Оператор требует от системы инициализировать регламент измерения. Система делает запись регламента измерения с указанием идентификатора создавшего Оператора, даты создания и статуса «Scheduled» (Намечен).

1.4. Завершение прецедента

Система подтверждает Оператору инициализацию регламента измерения и делает регламент измерения доступным для рассмотрения другими субъектами.

Описание прецедента. В этом стиле текст прост для чтения и поток событий легко проследить. Стремитесь к этому стилю в ваших описаниях.



Пример 2:

Операторы могут создавать Регламент сбора данных измерения элементов сети.

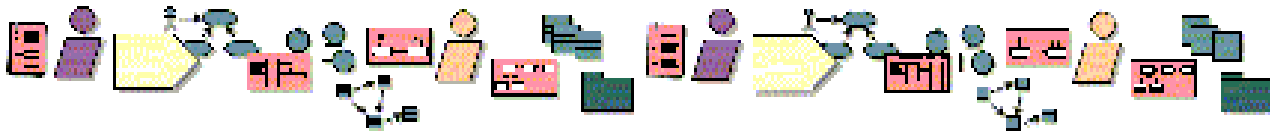
Система назначает Регламенту уникальное название и устанавливает значения по умолчанию, которые указывают длину и продолжительность измерения, а также как часто оно должно повторяться. Оператор должен иметь возможность редактировать эти значения.

Оператор должен далее определить, как соответствуют друг другу функции измерения, элементы сети и измеряемые объекты. Оператор может также добавлять к регламенту персональный комментарий.

Когда необходимая информация определена, создается и инициализируется новый Регламент с определенными атрибутами, идентификатором создателя и датой создания. Статусу регламента будет установлено значение «scheduled» (Намечен). Возможные значения статусов: Scheduled (Намечен), Executing (Выполняется), Completed (Закончен), Canceled (Отменен) и Erroneous (Ошибочный)

После этого интерфейс пользователя сообщает, что новый Регламент был создан и может быть отображен по запросу.

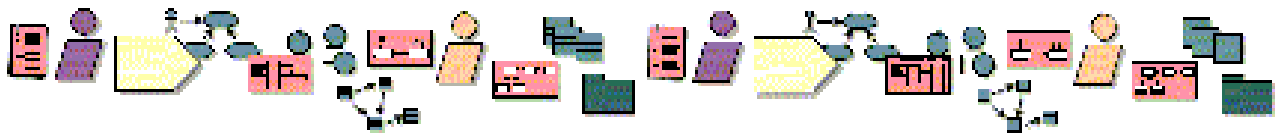
Описание прецедента: Этот стиль читаем, но не содержит ясного потока событий.



Пример 3:

```
"Администрирование регламента" (Идентификатор пользователя)
REPEAT
    <='Отображается меню администрирования регламента'
    IF (=> "Создание регламента" (Функция измерения,
        элемент сети, измеряемый объект)) THEN
        Система находит уникальное название и устанавливает по
        умолчанию когда и как долго должно выполняться измерение.
        <= "Отображается регламент" (Атрибуты по умолчанию)
        REPEAT
            =>'Редактирование регламента' (Назначение атрибутам
                новых значений)
            <= "Обновление экрана" (Новые атрибуты)
        UNTIL (Все атрибуты определены)
        REPEAT
            IF (=>'Редактирование регламента' (Назначение
                Атрибутам новых значений)
            THEN <= "Обновление экрана" (Новые атрибуты)
            ELSIF (=> "Сохранение регламента" (Идентификатор
                регламента, Атрибуты)) THEN
                Регламент создан и инициализирован в системе
                с определенными атрибутами, именем создателя,
                датой создания и статусом "scheduled" (Намечен).
                <= "Новый регламент создан" (Регламент)
            ENDIF
        UNTIL (=> "Завершение")
    ENDIF
UNTIL "Завершение администрирования регламента"
```

Описание прецедента: Здесь автор выбрал формальный стиль, используя псевдокод. Этот стиль делает понятными шаги процесса, и может быть полезен, если поток событий трудно описать словами.



Поток событий - пример

Законченное описание потока событий прецедента Управление регламентом, включая его альтернативные потоки, может выглядеть следующим образом:

1. Основной поток событий

1.1. Начало прецедента

Прецедент начинается, когда субъект Оператор требует от системы создать регламент измерения. Система ищет все элементы сети субъекта, их измеряемые объекты и соответствующие функции измерения, которые доступны этому конкретному Оператору. Доступные элементы сети – те, которые находятся в работе, и на обращение к которым Оператор имеет полномочия. Доступность функций измерения зависит от того, что было установлено для конкретного типа измеряемого объекта.

1.2. Конфигурирование регламента измерения

Система позволяет субъекту Оператор выбирать элементы сети для измерения, а затем показывает, какие измеряемые объекты доступны для выбранных элементов сети. Система позволяет Оператору выбрать измеряемые объекты, а затем выбрать, какие функции измерения устанавливать для каждого измеряемого объекта.

Система позволяет Оператору вводить текстовый комментарий относительно регламента измерения.

Оператор сообщает системе о завершении формирования регламента измерения. Система отвечает, генерируя уникальное название для регламента измерения и устанавливая значения по умолчанию для частоты и продолжительности измерения. Значения по умолчанию уникальны для каждого Оператора. Затем система позволяет Оператору редактировать эти значения по умолчанию.

1.3. Инициализация регламента

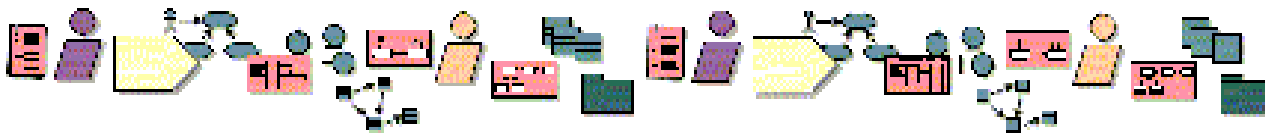
Оператор требует от системы инициализировать регламент измерения. Система делает запись регламента измерения с указанием идентификатора создавшего Оператора, даты создания и статуса «Scheduled» (Намечен).

1.4. Завершение прецедента

Система подтверждает Оператору инициализацию регламента измерения и делает регламент измерения доступным для рассмотрения другими субъектами.

2. Альтернативные потоки событий

2.1. Нет доступных элементов сети



Если на шаге 1.1, Начало прецедента, оказалось, что для этого Оператора сейчас нет доступных для измерения элементов сети, система сообщает об этом Оператору. После этого прецедент заканчивается.

2.2. Нет доступных функций измерения

Если на шаге 1.2, Конфигурирование регламента измерения, для выбранных элементов сети нет доступных функций измерения, система сообщит об этом Оператору и позволит Оператору выбирать другие элементы сети.

2.3. Отмена регламента измерения

Система позволит Оператору отменить все действия в любом пункте в процессе выполнения прецедента. В таком случае система возвратится в состояние, в котором она была до начала прецедента, и прецедент заканчивается.

Специальные требования

В специальных требованиях прецедента Вы описываете все требования прецедента, которые не отражаются потоком событий. Это требования, которые будут влиять на проектную модель, но не содержащие функциональности. Обсуждение требований, не содержащих функциональности, см. также на стр. 13-6. Вы можете организовывать эти требования в категориях типа Применимости, Надежности, Эффективности и Заменяемости, но обычно их так немного, что такая группировка не имеет особого значения.

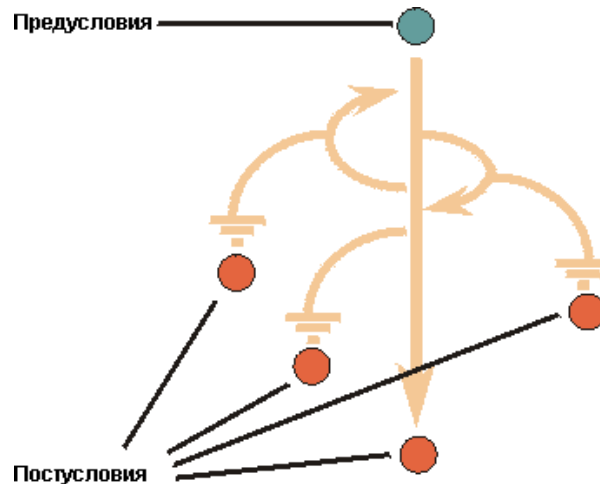
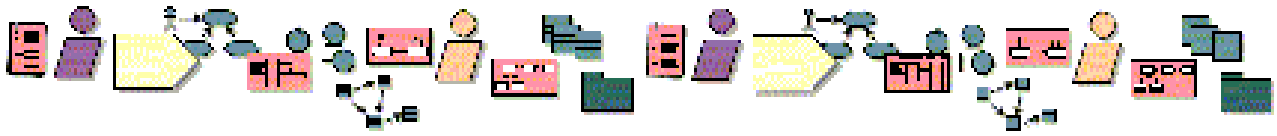
Пример:

В системе рециркуляторной машины специальным требованием прецедента Возврат предметов могло бы быть:

Машина должна быть способна опознавать предметы вложения с надежностью больше чем 95 процентов.

Предусловия и постусловия

Иногда для разъяснения того, как поток событий начинается и заканчивается, бывает полезно использовать понятия **предусловий** и **постусловий**. Однако, используйте их только, если это воспринимается аудиторией прецедента как полезная информация.



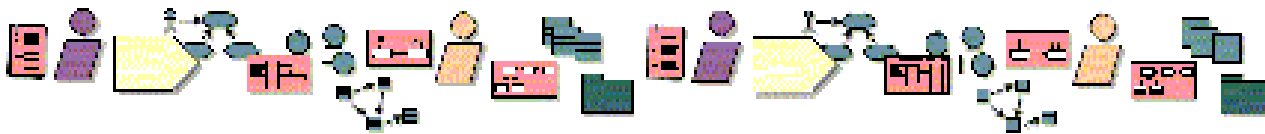
Предусловие – это состояние системы и ее окрестности, которое требуется прежде, чем прецедент может быть начат. Постусловие – это состояние, в котором будет система после того, как прецедент закончится.

Рассмотрите следующее:

- Состояния, описанные предусловиями или постусловиями, должны быть состояниями, которые пользователь может наблюдать. Состояния «Пользователь вошел в систему» или «Пользователь открыл документ» являются примерами наблюдаемых состояний.
- Предусловие – это ограничение для возможности начала прецедента. Прецедент начинается вовсе не событие.
- Предусловие для прецедента – это не предусловие только для одного подпотока, хотя Вы можете определять предусловия и постусловия и на уровне подпотоков.
- Постусловие для прецедента должно быть истинно независимо от того, который из альтернативных потоков был выполнен; оно не должно быть истинно только для главного потока. Если что-либо может потерпеть неудачу, Вы можете описать это в постусловии, говоря: «Действие закончено, или, если что-либо потерпело неудачу, действие не выполнено», а не только: «Действие закончено».
- Когда Вы используете постусловия вместе со связями расширения, Вы должны позаботиться, чтобы прецедент расширения не вводил подпоток, который нарушает постусловие в основном прецеденте.
- Постусловия могут быть мощным инструментом для описания прецедентов. Сначала Вы определяете то, чего прецедент, как предполагается, достигает, т.е. постусловия. Далее Вы можете описывать, как достичь этого состояния (необходимый поток событий).

Пример:

Предусловия для прецедента Снятие денег в системе банкомата: заказчик имеет



персонифицированную карту, которая соответствует считывающему устройству и зарегистрирована банковской системой.

Постусловие для прецедента Снятие денег в системе банкомата: в конце прецедента счет и данные транзакции сбалансированы, связь с банковской системой повторно инициализирована и заказчику возвращена его карта.

Точки расширения

Точка расширения открывает прецеденту возможность расширения. Она имеет название и список ссылок к одной или более точкам в пределах потока событий прецедента. Точка расширения может ссылаться на одно место между двумя шагами поведения в пределах прецедента. Или она может ссылаться на набор дискретных мест.

Использование названий точек расширения поможет Вам отделить технические требования поведения прецедента расширения от внутренних подробностей основного прецедента. Основной прецедент может изменяться или перестраиваться, но пока названия точек расширения остаются теми же, это не будет затрагивать прецедент расширения. В то же время, Вы не загружаете текст, описывающий поток событий основного прецедента, подробностями того, где его поведение могло бы быть расширено.

Пример:

В телефонной системе прецедент Телефонный вызов может быть расширен абстрактным прецедентом Показ вызывающего абонента. Это – необязательный сервис, часто упоминаемый как «Caller ID» (Идентификация номера телефона вызывающего абонента), который, возможно, потребуется стороной приема. Описание точки расширения в прецеденте Телефонный вызов может выглядеть следующим образом:

Название: Показ вызывающего абонента

Расположение: После раздела 1.9 Звонок телефона стороны приема.

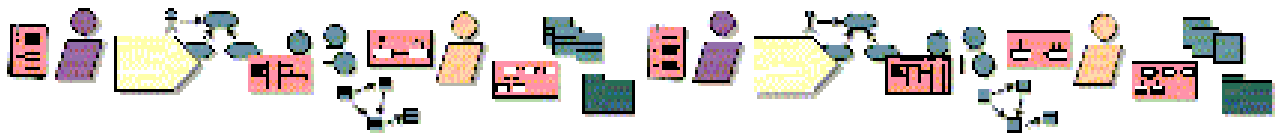
Диаграммы прецедентов

Вы можете проиллюстрировать связи прецедента с субъектами и с другими прецедентами с помощью диаграммы прецедентов (иногда – больше, чем одной диаграммы), принадлежащей прецеденту. Это полезно, если прецедент используется многими субъектами или имеет связи со многими другими прецедентами. Диаграмма этого вида имеет «локальный» характер, так как она показывает модель прецедентов с точки зрения только одного прецедента и не предназначена для объяснения общих фактов относительно всей модели прецедентов.

Диаграммы действий

Поток событий прецедента может быть описан графически с помощью диаграммы действий.

Мы подробно познакомились с диаграммами действий при описании основного потока работ Деловое моделирование (см. стр. 8-16, 9-11). Эти диаграммы показывают:



- Состояния действия, которые представляют выполняемое действие или шаг в пределах потока событий.
- Переходы, которые показывают последовательность действий. Этот тип перехода иногда называют переходом завершения, так как он отличается от перехода, в котором не требуется явного запуска события. Здесь событие запускается завершением действия, которое представляется состоянием действия.
- Решения, для которых определен набор условий защиты. Эта защита создает условия для контроля набора альтернативных переходов, которые следуют после окончания действия. Решения и условия защиты позволяют Вам показывать альтернативные потоки в потоке событий прецедента.
- Полосы синхронизации, которые Вы можете использовать, чтобы показать параллельные подпотоки. Полосы синхронизации позволяют Вам показывать конкурентные пути в потоке событий прецедента.

Артефакт: Пакет прецедентов

Пакет прецедентов – это совокупность прецедентов, субъектов, связей, диаграмм и других пакетов; он используется для структурирования модели прецедентов путем разделения ее на меньшие части.

Вы можете выделять разделы модели прецедентов в пакеты прецедентов по многим причинам:

- Вы можете использовать пакеты прецедентов для отражения очередности создания, конфигурации или поставки частей в законченной системе.
- Распределение ресурсов и компетентности различных групп разработки может потребовать, чтобы проект был разделен между ними. Некоторые пакеты прецедентов подходят для группы, а некоторые – для одного человека. Правильное разбиение позволяет разрабатывать пакеты наиболее эффективным способом. Вы должны убедиться, однако, что задачи каждого пакета определены так, что разработка может выполняться параллельно.
- Вы можете использовать пакеты прецедентов для структурирования модели прецедентов в соответствии с типами пользователей. Многие изменения требований происходят от пользователей. Пакеты прецедентов гарантируют, что изменения, поступившие от специфического типа пользователей, затронут только те части системы, которые соответствуют этому типу пользователей.
- В некоторых приложениях определенная часть информация должна иметь ограниченное распространение. Пакеты прецедентов позволяют Вам сохранять тайну в тех областях, где это необходимо.

В следующем, заключительном, выпуске, посвященном потоку работ Требования, мы обсудим артефакты, которые формирует Проектировщик интерфейса пользователя.

