**1. Образец\_прихода\_заявок**

Используемые ресурсы: «\_счетчик» (тип – «счетчики») и «заявка» (тип – «заявки»).

Условия выполнения паттерна: никаких особых условий не требуется.

В этом паттерне:

для ресурса «\_счетчик»

- показатель "количество\_заявок\_в\_очереди" увеличивается на единицу, это необходимо для статистики;

- показатель "счет\_заявок" увеличивается на единицу, это необходимо для статистики, а также для присваивания создаваемым заявкам уникальных номеров;

для ресурса «заявка»

- создается новая заявка;

- ей присваивается индивидуальный порядковый номер, передаваемый из ресурса «\_счетчик»;

- выбирается вес груза и маршрут (оба атрибута выбираются из соответствующих вероятностных распределений);

- заявке присваивается состояние "пришла", которое определит при работе модели использование ресурса заявка в том или ином паттерне.

**2. Образец\_расчета\_количества\_вагонов**

Используемые ресурсы: «заявка» (тип – «заявки»)

Условия выполнения паттерна: в этот паттерн может попасть только созданная (существующая) заявка, т.е. заявка в состоянии "пришла".

В этом паттерне:

- рассчитывается количество вагонов, которое необходимо изъять из парка вагонов для выполнения этой заявки;

- заявке присваивается новое состояние - "рассчитана", - определяющее при работе модели дальнейшее использование ресурса заявка в том или ином паттерне.

**3. Образец\_подготовки\_формируемого\_состава\_к\_обработке\_текущей\_заявки**

Используемые ресурсы: «заявка» (тип – «заявки») и группа ресурсов «формируемый\_состав» (тип – «формируемые составы»)

Условия выполнения паттерна: в этот паттерн может попасть только заявка в состоянии "рассчитана" (если рассматривается совершенно новая заявка) или "остались\_нераспределенные\_вагоны" (если часть вагонов из заявки уже помещена в формируемый\_состав), формируемый\_состав в состоянии "пустой" (если формируется новый формируемый\_состав) или "полупустой" (если в формируемом\_составе осталось место). Кроме того, маршрут в заявке и в формируемом\_составе должен совпадать, т.к. релевантных ресурсов типа «формируемые\_составы» в модели три и каждый формируется для определенного маршрута - для того, чтобы на каждом маршруте была своя очередь комплектования составов.

В этом паттерне:

для ресурса «заявка»

а) если заявка.состояние = рассчитана

- полю "остаток\_в\_заявке" присваивается значение "количество\_вагонов\_в\_заявке" - смысл этого присваивания состоит в том, что при комплектовании составов от общего количества вагонов в заявке будет "отщипываться" некоторое количество вагонов, которое может поместиться в текущий комплектуемый состав, но при этом рассчитанное количество\_вагонов\_в\_заявке изменяться не должно;

- состояние заявки изменяется на "привязать\_к\_формируемому\_составу", это состояние не позволит во время работы изменять ресурс «заявка» до тех пор, пока она не будет связана с конкретным формируемым\_составом;

б) если заявка.состояние = остались\_нераспределенные\_вагоны

- состояние заявки изменяется на "привязать\_к\_формируемому\_составу", это состояние не позволит во время работы изменять ресурс «заявка» до тех пор, пока он не будет связана с конкретным формируемым\_составом;

для группы ресурсов «формируемый\_состав» возможны два случая - формируемый\_состав в состоянии «пустой» и формируемый\_состав в состоянии «распределение\_вагонов» (это значит, что текущая заявка может быть распределена как в пустой (т.е. новый) формируемый\_состав, так и в наполняемый)

а) формируемый\_состав.состояние = пустой

- номер\_формируемого\_состава увеличивается на единицу, это значит, что начинает комплектоваться новый состав и ему присваевается очередной порядковый номер;

- состояние формируемого\_состава меняется на "привязать\_к\_следующей\_заявке", чтобы ресурс «формируемый\_состав» не изменялся до тех пор, пока не будет привязан к текущей заявке;

- в поле "номер\_распределяемой\_заявки" передается значение заявка.номер\_заявки - это необходимо для того, чтобы текущая заявка попала на распределение именно в этот формируемый\_состав, а не любой другой;

б) формируемый\_состав.состояние = полупустой

- в поле "номер\_распределяемой\_заявки" передается значение заявка.номер\_заявки - это необходимо для того, чтобы текущая заявка попала на распределение именно в этот формируемый\_состав, а не любой другой (при этом текущий номер распределяемой заявки у формируемого\_состава стирается, но это не страшно, т.к. невозможно сохранить у формируемого\_состава все номера распределенных в него заявок - непонятно, сколько нужно дополнительных атрибутов и как их отслеживать)

- состояние формируемого\_состава меняется на "привязать\_к\_следующей\_заявке", чтобы ресурс «формируемый\_состав» не изменялся до тех пор, пока не будет привязан к текущей заявке.

**4. Образец\_привязки\_состава\_к\_обрабатываемой\_заявке**

Используемые ресурсы: группа ресурсов «формируемый\_состав» (тип – «формируемые составы») и заявка (тип – «заявки»)

Условия выполнения паттерна: в паттерн попадает формируемый\_состав в состоянии "привязать\_к\_следующей\_заявке", номер которого совпадает с номером распределяемой заявки, а также заявка в состоянии "привязать\_к\_формируемому\_составу"

В этом паттерне:

для группы ресурсов «формируемый\_состав»

- состояние меняется на «распределение\_вагонов» - означает, что формируемый\_состав и заявка связаны и можно начинать наполнять состав вагонами из заявки;

для ресурса «заявка»

- в поле "номер\_формируемого\_состава1" передается значение "формируемый\_состав.номер\_распределяемой\_заявки" для того, чтобы при комплектации формируемого\_состава вагонами из заявок вагоны в него передавались из конкретной заявки

- состояние меняется на "распределение\_вагонов" - аналогично смене состояния формируемого\_состава в этом паттерне.

Примечание: образцах под номерами 3 и 4 происходит обмен номерами между заявкой и формируемым\_составом. В одном паттерне реализовать нельзя (или просто лично я не могу) из-за особенностей логики РДО.

**5. Образец\_распределения\_заявки\_заявка**

Используемые ресурсы: группа ресурсов «формируемый\_состав» (тип – «формируемые составы») и «заявка» (тип – «заявки»).

Условия выполнения паттерна: формируемый\_состав и заявка попадают в этот паттерн только в состоянии «распределение\_вагонов», должно выполняться соответствие их маршрутов и номеров

В этом паттерне проверяется наличие свободного места в составе:

для группы ресурсов «формируемый\_состав»

- ничего не меняется

для ресурса «заявка» возможны два случая:

а) свободного места в формируемом\_составе недостаточно для всех вагонов из заявки

- от заявки "отщипывается" количество вагонов, которое помещается в состав

- состояние меняется на "остались\_нераспределенные\_вагоны", чтобы эта заявка могла попасть в Образец\_подготовки\_формируемого\_состава\_к\_обработке\_текущей\_заявки для связывания с новым составом

б) свободного места достаточно

- поле остаток\_в\_заявке приобретает значение 0, это значит, что все вагоны из заявки распределены. Это справедливо и если остаток\_в\_заявке был равен количеству\_вагонов\_в\_заявке, и если остаток\_в\_заявке был меньше количества\_вагонов\_в\_заявке (т.е. часть вагонов укомплектована в предыдущий состав)

- состояние заявки меняется на "распределена", чтобы ограничить ее вхождение в различные образцы.

Примечание: Стоит заметить, что после обработки заявки в этом паттерне в поле "номер\_формируемого\_состава1" остается номер состава, в который ушли последние вагоны из заявки. В дальнейшем это позволит установить, когда заявка становится обслуженной.

**6. Образец\_распределения\_заявки\_формируемый\_состав**

Используемые ресурсы: «заявка» (тип – «заявки») и группа ресурсов «формируемый\_состав» (тип – «формируемые\_составы»)

Условия выполнения паттерна: те же, что и для паттерна 5.

В этом паттерне:

для ресурса «заявка»

- ничего не меняется;

для группы ресурсов «формируемый\_состав» возможны два случая:

а) в формируемом\_составе не хватает места для всех вагонов из заявки

- текущее\_количество\_вагонов = 60, это значит, что в свободное место в составе гарантированно заполнится

- состояние меняется на "создан", это значит, что формируемый\_состав наполнен и можно переходит к созданию реального состава.

б) в формируемом\_составе хватает места для всех вагонов из заявки

- к текущему\_количеству\_вагонов добавляются все вагоны из заявки

- состояние меняется на "полупустой", это значит, что в формируемом\_составе осталось место и он попадет в Образец\_подготовки\_формируемого\_состава\_к\_обработке\_текущей\_заявки для связывания с новой Заявкой.

Примечание: в паттерны 5 и 6 теоретически попадут связанные ресурсы формируемый\_состав и Заявка, поэтому формально можно считать, что оба паттерна представляют собой одно действие по изменению связанной пары ресурсов, разбитое на два паттерна из-за особенностей логики РДО.

**7. Образец\_готовности\_состава**

Используемые ресурсы: группа ресурсов «формируемый\_состав» (тип – «формируемые\_составы»), «состав» (тип – «составы») и «\_парк» (тип – «парки»).

Условия выполнения паттерна: в паттерн попадают только формируемые\_составы в состоянии «создан» - это главное и единственное условие для паттерна.

В этом паттерне:

для группы ресурсов «формируемый\_состав»

- обнуляется текущее\_количество\_вагонов, это необходимо потому, что фактически формируемый\_состав для каждого маршрута один; правомерно также обнулить этот атрибут в Образце\_подготовки\_формируемого\_состава\_к\_обработке\_текущей\_заявки;

- состояние меняется на «пустой», таким образом формируемый\_состав готов к распределению следующих заявок;

для ресурса «состав»

- создается новый состав;

- его состояние меняется на «сформирован», для того чтобы определить вхождение этого ресурса в конкретный образец;

- в поле «маршрут» передается значение аналогичного поля формируемого\_состава, чтобы во время работы модели состав «прошел» через контрольные точки соответствующего маршруту пути;

- в поле «номер\_состава» передается значение поля «формируемый\_состав.номер\_формируемого\_состава», чтобы отличать этот состав от других созданных составов;

- в поле «направление» передается значение «прямое», т.к. значение этого поля влияет на «продвижение» состава от пункта к пункту;

для ресурса «\_парк» возможны два случая

а) количества вагонов в парке хватает для формирования текущего состава

- из парка вычитается 60 вагонов;

б) количества вагонов в парке не хватает для формирования текущего состава

- количество\_вагонов\_в\_парке обнуляется, чтобы обозначить использование хотя бы имеющихся в наличии вагонов;

- в поле «дефицит» записывается размер нехватки вагонов.

**8. Образец\_создания\_состава**

Используемые ресурсы: «состав» (тип – «составы») и «\_станция\_отправления» (тип – «пункты\_отправления»)

Условия выполнения паттерна: в паттерн попадают только составы в состоянии «сформирован» при условии, что состояние ресурса \_станция\_отправления – «свободен».

В этом паттерне:

для ресурса «состав»

а) событие начала

- состояние состава меняется на «создается»;

б) событие конца

- состояние состава меняется на «готов\_к\_погрузке», чтобы определить вхождение ресурса в конкретный паттерн;

для ресурса «\_станция\_отправления»

а) событие начала

- состояние меняется на «занят», чтобы не происходило одновременное обслуживание нескольких составов;

б) событие конца

- состояние меняется на «свободен», что означает готовность ресурса к обслуживанию следующего состава.

**9. Образец\_погрузки**

Используемые ресурсы: «состав» (тип – «составы») и «\_пункт\_погрузки» (тип –«пункты\_погрузки»)

Условия выполнения паттерна: в паттерн попадают только составы в состоянии «сформирован» при условии, что состояние ресурса \_пункт\_погрузки – свободен.

В этом паттерне:

для ресурса «состав»

а) событие начала

- состояние состава меняется на «погрузка»;

б) событие конца

- состояние состава меняется на «готов\_к\_отправке», чтобы определить вхождение ресурса в конкретный паттерн;

для ресурса «\_пункт\_погрузки»

а) событие начала

- состояние меняется на «занят», чтобы не происходило одновременное обслуживание нескольких составов;

б) событие конца

- состояние меняется на «свободен», что означает готовность ресурса к обслуживанию следующего состава.

**10. Образец\_следования\_по\_маршруту**

Используемые ресурсы: «состав» (тип – «составы»)

Паттерн моделирует передвижение составов сразу по всем возможным участкам пути. Для этого у него заведены 5 параметров:

- номер\_маршрута, первый, второй или третий;

- начало\_пути, любой из всех возможных контрольных точек на всех маршрутах;

- конец\_пути, любой из всех возможных контрольных точек на всех маршрутах;

- расстояние, параметр, необходимый для расчета времени в пути;

- направление, параметр, определяющий направление движения состава.

Условие выполнения паттерна: состав должен находиться в состоянии «готов\_к\_отправке», его текущее местоположение должно совпадать со значением параметра «начало\_пути», его маршрут должен совпадать с параметром «маршрут» и направление его движения должно совпадать с параметром «направление».

В этом паттерне:

для ресурса «состав»

а) событие начала

- состояние меняется на «в\_пути», чтобы ресурс не был использован в других паттернах, пока не завершит движение;

б) событие конца

- состояние меняется на «прибыл», чтобы определить дальнейшие действия с составом;

- атрибут «местоположение» принимает значение параметра «конец\_пути», также для определения действий с составом.

**11. Образец\_определения\_действий\_с\_составом**

Используемые ресурсы: «станция\_отправления» (тип – «пункты\_отправления»), группа ресурсов «станция\_назначения» (тип – «пункты\_назначения»), «состав» (тип – «составы»).

Условия выполнения паттерна: в паттерн попадают только составы в состоянии «прибыл».

В этом паттерне:

для ресурса «станция\_отправления»

- ничего не меняется;

для группы ресурсов «станция\_назначения»

- ничего не меняется;

для ресурса «состав» возможны три случая:

а) состав прибыл на станцию\_назначения

- состояние меняется на «прибыл\_на\_разгрузку» для определения дальнейших действий с составом;

б) состав прибыл на станцию\_отправления

- состояние меняется на «прибыл\_в\_парк» для определения дальнейших действий с составом;

в) состав прибыл на таможню

- состояние меняется на «прибыл\_на\_таможню» для определения дальнейших действий с составом.

**12. Образец\_прохождения\_таможни**

Используемые ресурсы: группа ресурсов «таможня» (тип – «таможенные\_пункты»), «\_пункт\_смены\_колеи» (тип – «пункты\_смены\_колеи») и «состав» (тип – «составы»).

Условия выполнения паттерна: таможня должна находиться в состоянии «свободен», состав должен находиться в состоянии «прибыл\_на\_таможню».

В этом паттерне:

для группы ресурсов «таможня»

а) событие начала

- состояние меняется на «занят», чтобы не обслуживались одновременно несколько составов;

б) событие конца

- состояние меняется на «свободен», т.е. ресурс освобождается;

для ресурса «\_пункт\_смены\_колеи»

- ничего не происходит;

для ресурса «состав»

а) событие начала

- состояние меняется на «прохождение\_таможни», чтобы ресурс не использовался одновременно в нескольких образцах;

б) событие конца предполагает два варианта

1. состав находится в Бресте

- состояние меняется на «необходимо\_сменить\_колею», т.к. по условию в Бресте происходит замена колесных пар;

2. состав находится в любом другом таможенном пункте

- состояние меняется на «готов\_к\_отправке», что означает готовность ресурса вновь быть использованным в паттерне «Образец\_следования\_по\_маршруту».

**13. Образец\_смены\_колеи**

Используемые ресурсы: «состав» (тип – «составы»), группа ресурсов «\_пункт\_смены\_колеи» (тип – «пункты\_смены\_колеи»).

Условия выполнения паттерна: состав должен находиться в состоянии «необходимо\_сменить\_колею», \_пункт\_смены\_колеи должен находиться в состоянии «свободен».

В этом паттерне:

для ресурса «состав»

а) событие начала

- состояние меняется на «смена\_колеи», чтобы ресурс не был задействован в другом паттерне;

б) событие конца

- состояние меняется на «готов\_к\_отправке», что означает готовность состава следовать дальше по маршруту;

для группы ресурсов «\_пункт\_смены\_колеи»

а) событие начала

- состояние меняется на «занят», чтобы не происходило одновременного обслуживания нескольких составов;

б) событие конца

- состояние меняется на «свободен», что определяет готовность \_пункта\_смены\_колеи обслужить следующий состав.

**14. Образец\_разгрузки**

Используемые ресурсы: «состав» (тип – «составы»), группа ресурсов «пункт\_назначения» (тип – «пункты\_назначения»), «заявка» (тип – «заявки»), «\_счетчик» (тип – «счетчики»).

Условия выполнения паттерна: состав должен находиться в состоянии «прибыл\_на\_разгрузку», пункт\_назначения должен находиться в состоянии «свободен», местоположение состава и пункта\_назначения должны совпадать, номер состава, записанного в заявку в поле «номер\_формируемого\_состава1», должен совпадать с номером обрабатываемого состава.

В этом паттерне:

для ресурса «состав»

а) событие начала

- состояние меняется на «разгрузка»;

б) событие конца

- состояние меняется на «необходима\_мойка», чтобы определить вхождение ресурса в конкретный паттерн;

для ресурса «пункт\_назначения»

а) событие начала

- состояние меняется на «занят», чтобы не происходило одновременного обслуживания нескольких составов;

б) событие конца

- состояние меняется на «свободен», чтобы обозначить готовность пункта к обслуживанию следующего состава;

для ресурса «заявка»

а) событие начала

- отсутствует;

б) событие конца

- состояние меняется на «обслужена», т.к. весь груз из заявки выгружен на соответствующем пункте\_назначения;

для ресурса «\_счетчик»

а) событие начала

- отсутствует;

б) событие конца

- поле «количество\_обслуженных\_заявок» увеличивается на 1.

**15. Образец\_мойки**

Используемые ресурсы: «состав» (тип – «составы»), группа ресурсов «пункт\_мойки» (тип – «пункты\_мойки»).

Условия выполнения паттерна: состав должен находиться в состоянии «необходима\_мойка», пункт\_мойки должен находиться в состоянии «свободен», местоположение состава и пункта\_мойки должно совпадать.

В этом паттерне:

для ресурса «состав»

а) событие начала

- состояние меняется на «мойка»;

б) событие конца

- состояние меняется на «готов\_к\_возврату»;

- направление принимает значение «обратное», что определяет следующий пункт, в который попадет состав4

для ресурса «пункт\_мойки»

а) событие начала

- состояние меняется на «занят», чтобы не происходило одновременное обслуживание нескольких составов;

б) событие конца

- состояние меняется на «свободен».

**16. Образец\_возврата\_вагонов\_в\_парк**

Используемые ресурсы: «состав» (тип – «составы»), «парк» (тип – «парки»).

Условия выполнения паттерна: состав должен находиться в состоянии «прибыл\_в\_парк».

В этом паттерне:

для ресурса «состав»

- ничего не происходит, после продвижения модельного времени ресурс стирается;

для ресурса «парк»

а) событие начала

- отсутствует;

б) событие конца

- количество\_вагонов\_в\_парке увеличивается на 60, это значит, что состав вернулся в парк.

Примечание: в паттернах 3, 4, 5, 6, 7, 11, 12, 14, 15 употреблено понятие «группа ресурсов». Имеется в виду, что в этом паттерне может использоваться один из идентичных по набору атрибутов ресурс, отличающийся от других только значениями атрибутов. Использование конкретного ресурса из группы определяется условиями выполнения паттерна.

Примечание: смысл паттернов 8, 9, 11, 12, 13, 14, 15, 16 заключается только в продвижении модельного времени на время обслуживания.