**1. Образец\_очереди\_заявок**

Используются два ресурса: Заявка (тип - заявки) и \_система (тип - система - счетчик чего угодно при работе модели)

Условия выполнения паттерна: никаких особых условий не требуется.

В этом паттерне:

для ресурса \_система

- 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 \_система (которая в модели играет роль счетчика изменений ресурсов);

- 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**

Используется один ресурс: 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**

Используются два ресурса: Заявка (тип - заявки) и 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 она не будет связана с конкретным формируемым\_составом;

для ресурса формируемый\_состав возможны два случая - формируемый\_состав.пустой и формируемый\_состав.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**

Используются два ресурса: 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 ресурса формируемый\_состав

- состояние меняется на "распределение\_вагонов" - означает, что 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**

Используются два ресурса: 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 ресурса 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**

Используются два ресурса: Заявка (тип - заявки) и 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 ресурса 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**

Используются три ресурса: 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 ресурса 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 (ресурс состав создается, подобно ресурсу заявка)

- 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**

Используются два ресурса: состав (тип – составы) и \_станция\_отправления (тип – контрольные пункты)

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.

**8. Образец\_погрузки**

Используются два ресурса: состав (тип – составы) и \_пункт\_погрузки (тип – контрольные пункты)

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.

Примечание: смысл паттернов 7 и 8 только в продвижении модельного времени.