

## Лекция №5

### Раздел 2. Модельная динамика систем с дискретными событиями

Существует несколько концепций дискретно-событийного моделирования (ДСМ), определяющих различные подходы к декомпозиции динамических процессов, протекающих в моделируемой системе.

1. Концепция событий.
2. Концепция состояний.
3. Концепция параллельных процессов.

#### 2 Концепция событий в дискретно-событийном моделировании

##### 2.1 Структура календаря событий

При использовании концепции событий необходимо описать моделируемую систему множеством типов событий  $\{EV_1, EV_2, \dots, EV_K\}$  и установить закономерности (в том числе, вероятностные) возникновения событий каждого типа во времени, так как поток событий является хронологическим. Имитация процесса функционирования системы при этом связана с возникновением событий определенных типов и выполнением работ, соответствующих событиям различных типов, т.е. установлением реакции системы на эти события. Процесс возникновения событий имеет в общем случае стохастический характер, а моменты возникновения событий образуют хронологические потоки.

Динамикой работы моделируемой системы управляет специальная управляющая программа, называемая монитором моделирования или симулятором. Монитор выполняет упорядочение событий по времени их возникновения в системе, пересчет системного времени, а также слежение за выполнением условий активации работ. Для реализации и поддержки хронологических отношений между событиями в мониторах моделирования используется специальная информационная структура списка, называемая календарем событий (Рисунок 6). Календарь – это хронологически упорядоченное расписание, определяющее план развития модели во времени.

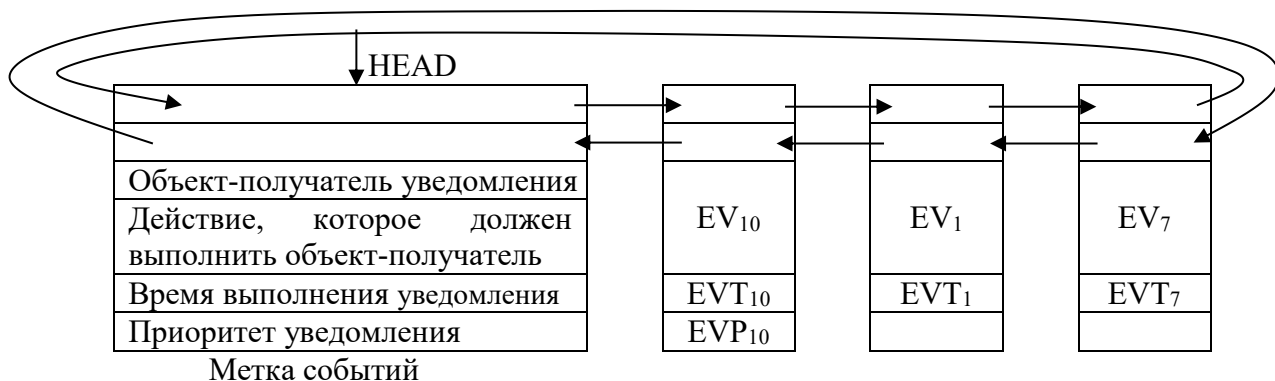


Рисунок 6 – Структура календаря событий

Календарь событий состоит из меток событий. Метка  $L$  события  $EV$  включает уведомление о типе события  $EV_k$ , где  $k$  – тип события, время возникновения события в системе (время выполнения уведомления)  $EVT_k$ , приоритет события  $EVP_k$  (не обязательно) и дополнительную информацию (не обязательно). Уведомление о типе события включает объект-получатель уведомления и действие, которое он должен выполнить. голове календаря (Рисунок 6), находится метка события 10-го типа

$$L(HEAD) = \langle EV_{10}, EVT_{10}, EVP_{10}, \text{дополнительная информация} \rangle.$$

Метка, стоящая в голове календаря, имеет минимальное время наступления. Метки в календаре упорядочены по времени наступления событий

$$EVT_{10} \leq EVT_1 \leq EVT_7.$$

В процессе моделирования календарь постоянно обновляется. В него вносятся метки новых событий и удаляются метки событий, соответствующие текущему времени, а также те события, которое в соответствии с логикой работы системы необходимо отменить. Хронологическое упорядочение календаря при этом должно сохраняться. Интерпретацию уведомления события, стоящего в голове календаря, можно определить, как наступление события, которое заключается в послышке сообщения некоторому объекту, где указывается, какое действие объект должен выполнить. В ответ объект-получатель уведомления выполняет активность (программу работ), соответствующую данному типу события. Событие является мгновенным, т.е. не имеет длительности в модельном времени, а программа работ (активность) характеризуется временем начала и окончания, т.е. имеет длительность в модельном времени.

Реализация календаря выполняется на структуре динамических списков. Наиболее эффективными для построения упорядоченных списков являются двунаправленные (двусвязные) циклические списки.

## 2.2 Два вида управления в дискретно-событийных моделях

В дискретно-событийном моделировании рассматриваются два вида событий: условные и безусловные. Безусловные события используются в моделях хронологических потоков. Безусловное событие выполняется точно в тот момент времени, когда запланированное время его наступления становится равно текущему значению модельного времени. Наступление условного события зависит не только от времени, но и от дополнительных условий.

В соответствии с двумя видами событий в концепции событий различают два вида управления: императивное и интеррогативное.

**Императивное управление** («imperative» – повелительное, указывающее) связано с управлением движением ленты фреймов и проверкой содержимого окна текущего времени. Операторы императивного управления планируют события на будущие моменты времени, включают метки этих событий в календарь, удаляют метки событий из календаря.

Операторы императивного управления могут явно указывать моменты времени:

- «Запланировать наступление события  $E V_K$  в момент времени  $T$ » – планирование в *абсолютном* времени.
- «Запланировать наступления события  $E V_K$  с задержкой на  $t$  единиц от текущего значения модельного времени» – планирование в *относительном* времени.

Операторы императивного управления могут не использовать значение времени в явном виде, а реализовывать механизм приоритетов:

- «Включить событие  $E V_K$  перед (после) события  $E V_n$ ».

Императивное управление всегда связано с изменением календаря событий и поддержанием хода часов модельного времени.

**Интеррогативное управление** («interrogative» – вопросительное) определяет процесс функционирования системы при достижении ею определенного состояния (статуса), т.е. при выполнении определенного набора логических условий. Интеррогативное управление не связано с ходом часов модельного времени. Операторы

интеррогативного управления постоянно следят за состоянием системы и обеспечивают реакцию на его изменение. Интеррогативное управление, в отличие от императивного, используется, когда невозможно заранее предсказать наступление события, не только детерминированным образом, но даже с использованием стохастических методов.

### 2.3 Схема интеррогативного управления

Между множеством типов событий  $\{EV_n, EV_m, \dots, EV_k\}$  и множеством логических условий  $\{CN_1, CN_2, \dots, CN_L\}$  устанавливается соответствие либо перед началом моделирования, либо в процессе моделирования.

Условие – это булева функция, определяемая набором логических условий, при выполнении которых система должна изменить свое состояние, при этом в системе наступает событие. Если условия выполнены, соответствующая функция  $CN$  активна. В процессе имитации после окончания выполнения программы работ очередного события управление передается монитору моделирования, который проверяет соответствие логического условия типу события ( $CN \leftrightarrow EV$ ), т.е. должно ли привести выполнение логических условий, определяющих функцию  $CN$ , к возникновению события  $EV$ ? Если соответствие имеется, проверяется активность функции  $CN$  ( $CN == true?$ ). Если функция  $CN$  активна, монитор передает управление программе работ соответствующего события. Если из всего множества функций нет ни одной активной, происходит пересчет системного времени и интеррогативное управление запускается снова. Схема интеррогативного управления показана на Рисунке 7.

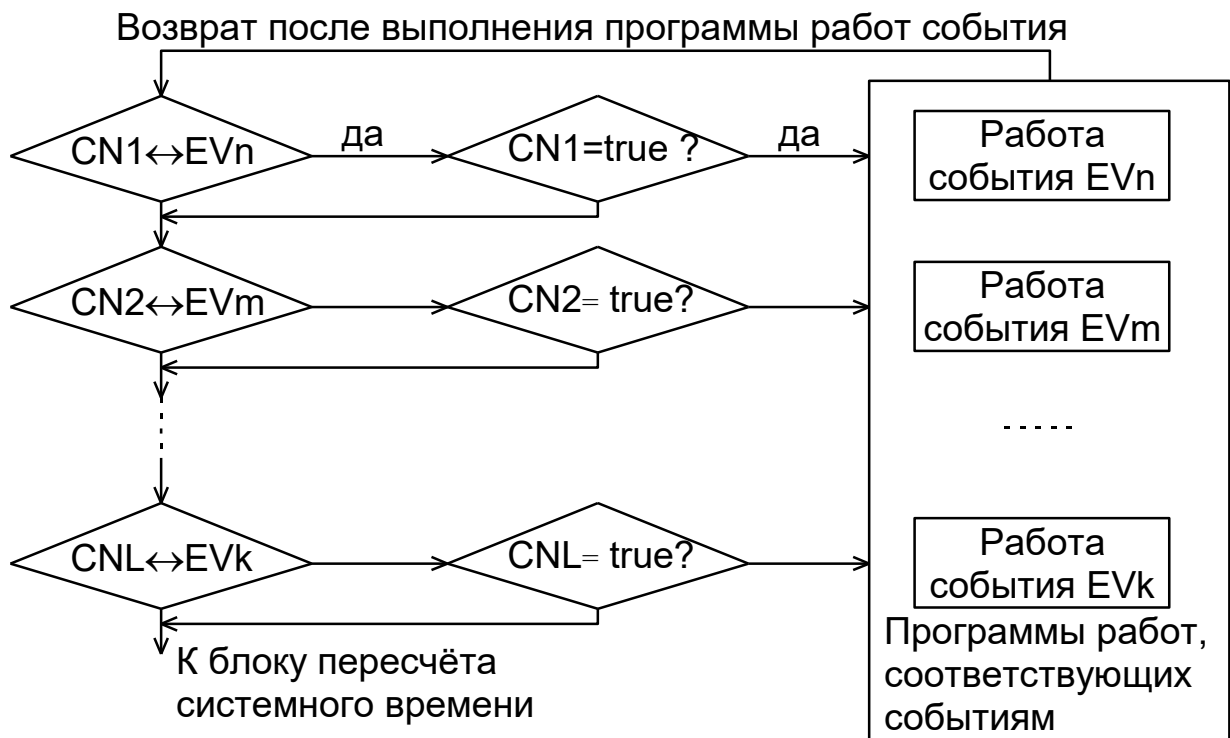


Рисунок 7 – Схема интеррогативного управления

Один тип события может быть связан с несколькими логическими условиями. Выполнение одного логического условия может вызвать наступление событий нескольких типов.

Программы работ условных событий и булевы функции условий работают с одним и тем же множеством разделяемых переменных, значения которых определяют состояние системы. Функции условий проверяют значения этих переменных, а программы работ изменяют их значения (Рисунок 8).

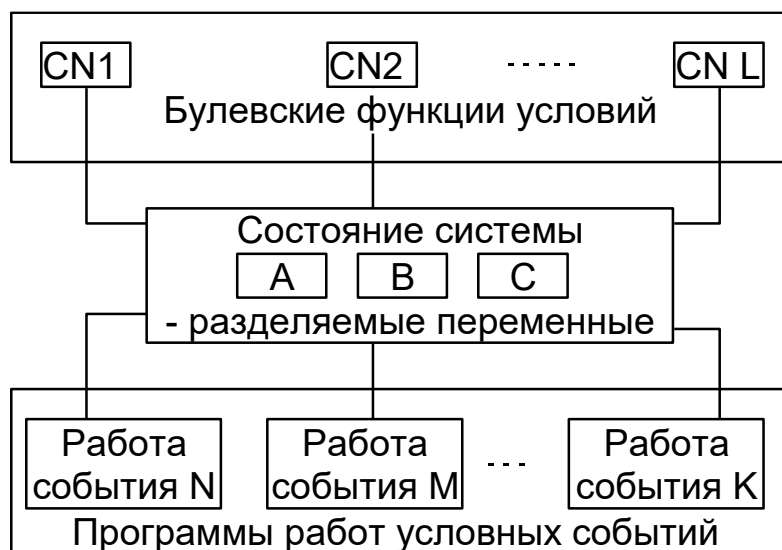
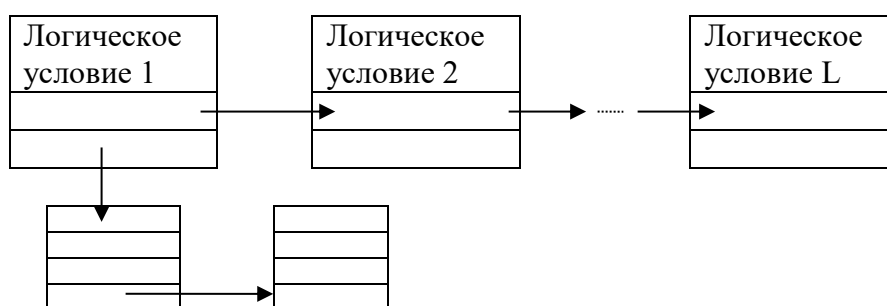


Рисунок 8 – Разделяемые переменные в схеме интеррогативного управления

Основные операторы интеррогативного управления:

- «Запланировать наступление события  $EV_k$  в связи с выполнением функции условий  $CN_L$ ». Если  $CN_L$  новое условие, монитор моделирования должен включить новый элемент в множество условий.
- «Разрушить связь между событием  $EV_k$  и условием  $CN_L$ ». Если условие  $CN_L$  не связано с другими типами событий, монитор моделирования исключает его из множества логических условий.

Интеррогативное управление использует информационную структуру мультисписка следующего вида (Рисунок 9).



Список событий, ожидающих выполнения условия

Рисунок 9 – Структура мультисписка для реализации интеррогативного управления

Схема интеррогативного управления может быть описана следующей циклической структурой:

WHILE <множество логических условий  $\{CN\}$  содержит хотя бы один активный элемент, т.е. существует функция  $(CN == true)$  в текущий момент времени>

DO <выполнить программы работ всех событий, связанных со всеми активными условиями>.

Операторы интеррогативного управления не изменяют содержимое календаря событий и не управляют ходом часов модельного времени. При интеррогативном управлении модельное время останавливается, при этом имитируются процессы, происходящие в модельном времени мгновенно. В компьютерном времени цикл интеррогативного управления может выполняться определенное время, т.к. программы работ событий могут изменить состояние системы и, следовательно, множество логических условий  $\{CN\}$ , что приведет к новому запуску цикла интеррогативного управления.

## 2.4 Схема универсального дискретно-событийного симулятора

Универсальный дискретно-событийный симулятор попеременно выполняет функции императивного и интеррогативного управления на всем протяжении имитационного эксперимента.

```
WHILE <имитационный эксперимент продолжается>
DO
    (*интеррогативное управление*)
    WHILE <множество  $\{CN\}$  содержит хотя бы один активный элемент, т.е.
        существует функция  $(CN==true)$  в текущий момент времени>
    DO <выполнить программы работ всех событий, связанных со всеми
        активными условиями>
    END;
    (*императивное управление*)
    переместить ленту фреймов влево так, чтобы непустой фрейм попал в
    окно текущего времени, т.е. выполнить пересчет системного времени;
    выбрать метки всех событий, находящиеся в окне текущего времени;
    выполнить программы работ событий, соответствующих этим меткам;
END;
```