

# Программное моделирование информационных систем

Использование современных ЭВМ, вычислительных комплексов и сетей является мощным средством реализации имитационных моделей и исследования с их помощью характеристик процесса функционирования систем  $S$ . Эффективность исследования системы  $S$  на программно-реализуемой модели  $M_m$  прежде всего зависит от правильности схемы моделирующего алгоритма, совершенства программы и только косвенным образом зависит от технических характеристик ЭВМ, применяемой для моделирования. Большое значение при реализации модели на ЭВМ имеет вопрос правильного выбора языка моделирования.

**Алгоритмические языки** при моделировании систем служат вспомогательным аппаратом разработки, машинной реализации и анализа характеристик моделей. Каждый язык моделирования должен отражать определенную структуру понятий для описания широкого класса явлений.

Основными моментами, характеризующими качество языков моделирования, являются: удобство описания процесса функционирования системы  $S$ , удобство ввода исходных данных моделирования и варьирования структуры, алгоритмов и параметров модели, реализуемость статистического моделирования, эффективность анализа и вывода результатов моделирования, простота отладки и контроля работы моделирующей программы, доступность восприятия и использования языка.

**Язык программирования** представляет собой набор символов, распознаваемых ЭВМ и обозначающих операции, которые можно реализовать на ЭВМ. На низшем уровне находится основной язык машины, программа на котором пишется в кодах, непосредственно соответствующих элементарным машинным действиям (сложение, запоминание, пересылка по заданному адресу и т. д.). Следующий уровень занимает автокод (язык **АССЕМБЛЕРА**) вычислительной машины. Программа на автокоде составляется из мнемонических символов, преобразуемых в машинные коды специальной программой — ассемблером.

**Компилятором** называется программа, принимающая инструкции, написанные на алгоритмическом языке высокого уровня, и преобразующая их в программы на основном языке машины или на автокоде, которые в последнем случае транслируются еще раз с помощью ассемблера.

**Интерпретатором** называется программа, которая, принимая инструкции входного языка, сразу выполняет соответствующие операции в отличие от компилятора, преобразующего эти инструкции в запоминающиеся цепочки команд. Трансляция происходит в течение всего времени работы программы, написанной на языке интерпретатора. В отличие от этого компиляция и ассемблирование представляют собой однократные акты перевода текста с входного языка на объектный язык машины, после чего полученные программы выполняются без повторных обращений к транслятору.

Программа, составленная в машинных кодах или на языке **АССЕМБЛЕРА**, всегда отражает специфику конкретной ЭВМ.

Инструкции такой программы соответствуют определенным машинным операциям и, следовательно, имеют смысл только в той ЭВМ, для которой они предназначены, поэтому такие языки называются **машинно-ориентированными языками**.

Большинство языков интерпретаторов и компиляторов можно классифицировать как **процедурно-ориентированные языки**. Эти языки качественно отличаются от **машинно-ориентированных языков**, описывающих элементарные действия ЭВМ и не обладающих проблемной ориентацией.

Все **процедурно-ориентированные языки** предназначены для определенного класса задач, включают в себя инструкции, удобные для формулировки способов решения типичных задач этого класса. Соответствующие алгоритмы программируются в обозначениях, не связанных ни с какой ЭВМ.

Язык моделирования представляет собой процедурно-ориентированный язык, обладающий специфическими чертами. Основные языки моделирования разрабатывались в качестве программного обеспечения имитационного подхода к изучению процесса функционирования определенного класса систем.

## Особенности использования алгоритмических языков

Рассмотрим преимущества и недостатки использования для моделирования процесса функционирования систем **языков имитационного моделирования (ЯИМ)** и **языков общего назначения (ЯОН)**, т. е. универсальных и процедурно-ориентированных алгоритмических языков.

Целесообразность использования **языков имитационного моделирования (ЯИМ)** вытекает из двух основных причин:

**1) удобство программирования модели системы, играющее существенную роль при машинной реализации моделирующих алгоритмов;**

**2) концептуальная направленность языка на класс систем, необходимая на этапе построения модели системы и выборе общего направления исследований в планируемом машинном эксперименте.**

Практика моделирования систем показывает, что именно использование ЯИМ во многом определило успех имитации как метода экспериментального исследования сложных реальных объектов.

Языки моделирования позволяют описывать моделируемые системы в терминах, разработанных на базе основных понятий имитации. До того как эти понятия были четко определены и формализованы в ЯИМ, не существовало единых способов описания имитационных задач, а без них не было связи между различными разработками в области постановки имитационных экспериментов. Высокоуровневые языки моделирования являются удобным средством общения заказчика и разработчика машинной модели  $M_m$ .

Несмотря на перечисленные преимущества ЯИМ, в настоящее время выдвигаются основательные аргументы как технического, так и эксплуатационного характера против полного отказа при моделировании от универсальных и процедурно-ориентированных языков.

Серьезные недостатки ЯИМ проявляются в том, что в отличие от широко применяемых ЯОН, трансляторы с которых включены в поставляемое изготовителем математическое

обеспечение всех современных ЭВМ, языки моделирования, за небольшим исключением, разрабатывались отдельными организациями для своих достаточно узко специализированных потребностей. Соответствующие трансляторы плохо описаны и приспособлены для эксплуатации при решении задач моделирования систем, поэтому, несмотря на достоинства ЯИМ, приходится отказываться от их практического применения в ряде конкретных случаев.

При создании системы моделирования на базе любого языка необходимо решить вопрос о синхронизации процессов в модели, так как в каждый момент времени, протекающего в системе (системного времени), может потребоваться обработка нескольких событий, т. е. требуется псевдопараллельная организация имитируемых процессов в машинной модели  $M_m$ . Это является основной задачей монитора моделирования, который выполняет следующие функции: управление процессами (согласование системного и машинного времени) и управление ресурсами (выбор и распределение в модели ограниченных средств моделирующей системы).

## **Подходы к разработке языков моделирования**

К настоящему времени сложились два различных подхода к разработке языков моделирования: непрерывный и дискретный — отражающие основные особенности исследуемых методом моделирования систем. Поэтому ЯИМ делятся на две самостоятельные группы, которые соответствуют двум видам имитации, развивавшимся независимо друг от друга: для имитации непрерывных и дискретных процессов.

Для моделирования непрерывных процессов могут быть использованы не только АВМ, но и ЭВМ, последние при соответствующем программировании имитируют различные непрерывные процессы. При этом ЭВМ обладают большей надежностью в эксплуатации и позволяют получить высокую точность результатов, что привело к разработке языков моделирования, отображающих модель в виде блоков таких типов, которые играют роль стандартных блоков АВМ (усилителей, интеграторов, генераторов функций и т. п.).

Заданная схема моделирующего алгоритма преобразуется в систему совместно рассматриваемых дифференциальных уравнений. Моделирование в этом случае сводится, по сути дела, к отысканию численных решений этих уравнений при использовании некоторого стандартного пошагового метода.

**Универсальная ЭВМ** — устройство дискретного типа, а поэтому должна обеспечивать дискретную аппроксимацию процесса функционирования исследуемой системы  $S$ . Непрерывные изменения в процессе функционирования реальной системы отображаются в дискретной модели  $M_u$ , реализуемой на ЭВМ, некоторой последовательностью дискретных событий, и такие модели называются **моделями дискретных событий**. Отдельные события, отражаемые в дискретной модели, могут определяться с большой степенью приближения к действительности, что обеспечивает адекватность таких дискретных моделей реальным процессам, протекающим в системах  $S$ .

**Архитектуру ЯИМ**, т. е. концепцию взаимосвязей элементов языка как сложной системы, и технологию перехода от системы  $S$

к ее машинной модели  $M_b$ , можно представить следующим образом:

- 1)** объекты моделирования (системы  $S$ ) описываются (отображаются в языке) с помощью некоторых атрибутов языка;
- 2)** атрибуты взаимодействуют с процессами, адекватными реально протекающим явлениям в моделируемой системе  $S$ ;
- 3)** процессы требуют конкретных условий, определяющих логическую основу и последовательность взаимодействия этих процессов во времени;
- 4)** условия влияют на события, имеющие место внутри объекта моделирования (системы  $S$ ) и при взаимодействии с внешней средой  $E$ ;
- 5)** события изменяют состояния модели системы  $M$  в пространстве и во времени.

В большинстве случаев с помощью машинных моделей исследуются характеристики и поведение системы  $S$  на определенном отрезке времени, поэтому одной из наиболее важных задач при создании модели системы и выборе языка программирования модели является реализация двух функций:

- 1) корректировка временной координаты состояния системы («продвижение» времени, организация «часов»);
- 2) обеспечение согласованности различных блоков и событий в системе (синхронизация во времени, координация с другими блоками).

## **Классификации языков моделирования**

Для машинного моделирования системы  $S$  пригодны три способа проведения вычислений, в основе которых лежит

применение цифровой, аналоговой и гибридной вычислительной техники рис. 1.

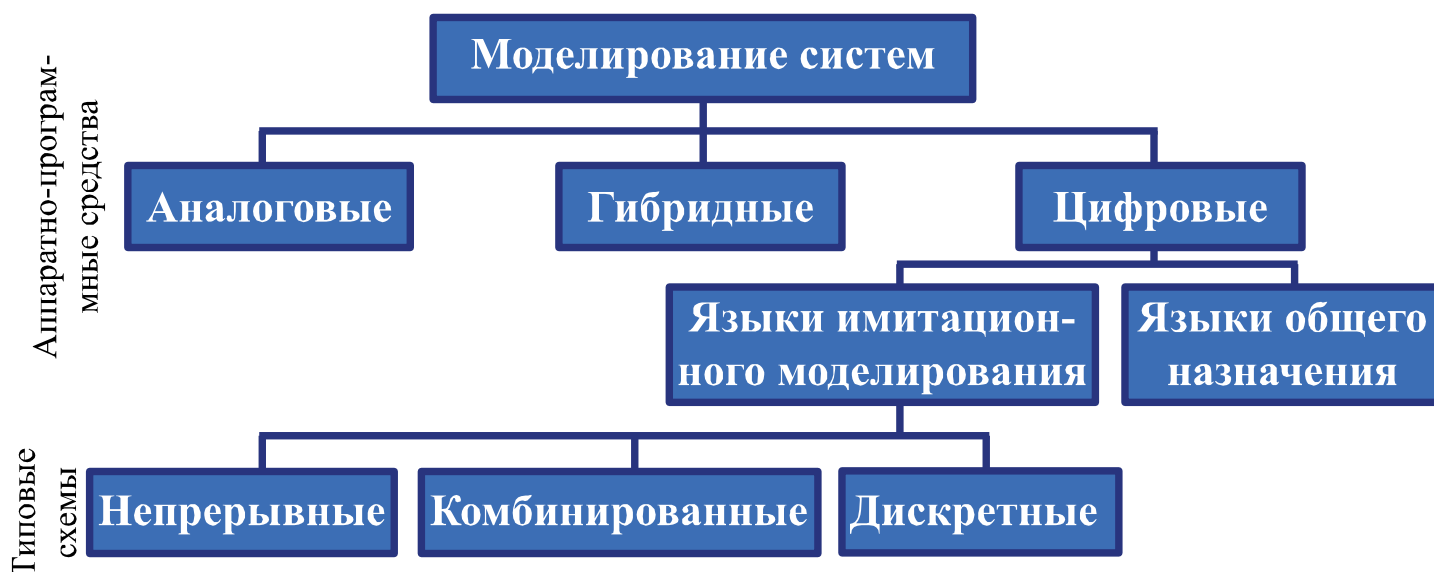


Рис. 1.

Для моделирования систем используются как универсальные и процедурно-ориентированные ЯОН, так и специализированные ЯИМ. При этом ЯОН предоставляют программисту-разработчику модели  $M_m$  больше возможностей в смысле гибкости разработки, отладки и использования модели. Но гибкость приобретается ценой больших усилий, затрачиваемых на программирование модели, так как организация выполнения операций, отсчет системного времени и контроль хода вычислений существенно усложняются.

Имеющиеся ЯИМ можно разбить на три основные группы, соответствующие трем типам математических схем: непрерывные, дискретные и комбинированные. Языки каждой группы предназначены для соответствующего представления системы  $S$  при создании ее машинной модели  $M_m$ .

В основе рассматриваемой классификации в некоторых ЯИМ лежит принцип формирования **системного времени**. Так как «системные часы» предназначены не только для продвижения



системного времени в модели  $M_m$ , но также для синхронизации различных событий и операций в модели системы  $S$ , то при отнесении того или иного конкретного языка моделирования к определенному типу нельзя не считаться с типом механизма «системных часов».

**Непрерывное** представление системы  $S$  сводится к составлению уравнений, с помощью которых устанавливается связь между зависимыми и независимыми переменными модели. Примером такого непрерывного подхода является использование дифференциальных уравнений. Причем в дальнейшем дифференциальные уравнения могут быть применены для непосредственного получения характеристик системы.

Представление системы  $S$  в виде типовой схемы, в которой участвуют как непрерывные, так и дискретные величины, называется **комбинированным**. Состояние модели системы  $M(S)$  описывается набором переменных, некоторые из которых меняются во времени непрерывно. Законы изменения непрерывных компонент заложены в структуру, объединяющую дифференциальные уравнения и условия относительно переменных. Предполагается, что в системе могут наступать события двух типов:

- 1) события, зависящие от состояния системы;
- 2) события, зависящие от времени.

События первого типа наступают в результате выполнения условий, относящихся к законам изменения непрерывных переменных.

Для событий второго типа процесс моделирования состоит в продвижении системного времени от момента наступления события до следующего аналогичного момента.

В рамках **дискретного подхода** можно выделить несколько принципиально различных групп ЯИМ.

Первая группа ЯИМ подразумевает наличие списка событий, отличающих моменты начала выполнения операций. Продвижение времени осуществляется по событиям, в моменты наступления которых производятся необходимые операции, включая операции пополнения списка событий.

При использовании ЯИМ второй группы после пересчета системного времени, в отличие от схемы языка событий, просмотр действий с целью проверки выполнения условий начала или окончания какого-либо действия производится непрерывно. Просмотр действий определяет очередность появления событий.

Третья группа ЯИМ описывает системы, поведение которых определяется процессами. В данном случае под процессом понимается последовательность событий, связь между которыми устанавливается с помощью набора специальных отношений. Динамика заложена в независимо управляемых программах, которые в совокупности составляют программу процесса.