

ТЕМА 3. Инструментальные средства имитационного моделирования

Введение

Использование AnyLogic предоставляет уникальную возможность войти в мир моделирования, имея лишь базовую подготовку в области информационных технологий.

Это современная среда разработки моделей на языке Java с русскоязычным графическим интерфейсом и тщательно продуманной контекстной справочной системой. AnyLogic содержит большую библиотеку визуальных компонентов. Разработчик может также создавать и добавлять в среду собственные компоненты. Модели сохраняются как Java-апплеты. В профессиональной версии работает отладчик и можно создавать автономные JAR-файлы. AnyLogic-модели обладают хорошими средствами 2D–3D симуляции, интерактивности и развитыми возможностями проведения экспериментов (в том числе оптимизационных).

После установки программы нужно пройти регистрацию на сайте компании и получить ознакомительный (бесплатный) или постоянный ключ. Регистрация с постоянным ключом дает возможность получать от группы поддержки помощь в разработке моделей по Интернету, а также обновления программы. Необходимо отметить, что разработчики регулярно обновляют учебную версию AnyLogic, сближая ее возможности с профессиональной версией более ранних релизов.

При запуске AnyLogic отображается начальная страница (рис. 1). После этого можно выбрать пример из обширного списка учебных моделей и посмотреть ее работу.

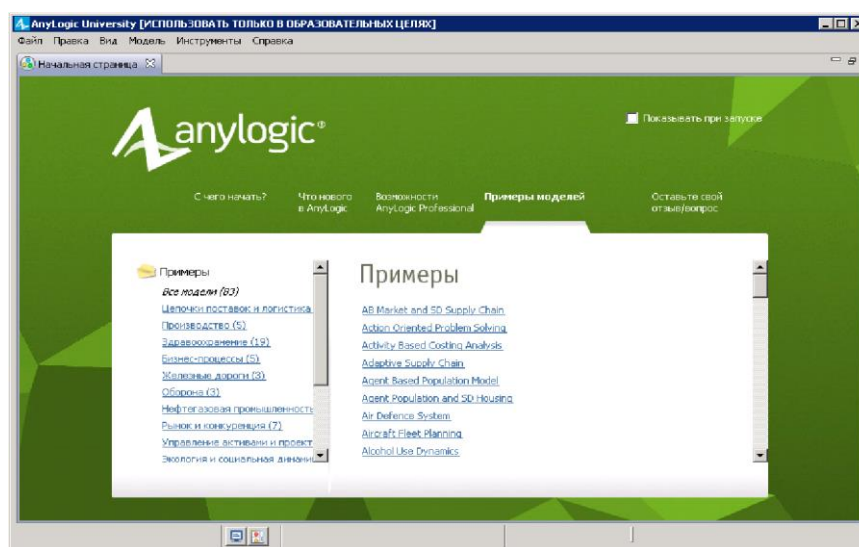


Рис.1. Стартовая страница AnyLogic

Другой вариант – закрыть стартовую страницу и начать работу «с нуля». Перед этим можно снять флажок у метки *Показывать при запуске*, чтобы в следующий раз сразу же начинать работу с моделью.

1. Структура программной среды и базовые элементы программы имитационного моделирования Any Logic.

Окно AnyLogic в начале работы над новой моделью показано на рис. 2.2, где в верхней части находится строка меню и панели инструментов. В центре расположена область построения модели – *Графический редактор* (вкладка *Main*). По периферии – рабочие панели среды моделирования с соответствующими вкладками и кнопками управления размерами. Их можно вынести за пределы главного окна AnyLogic или закрыть совсем.

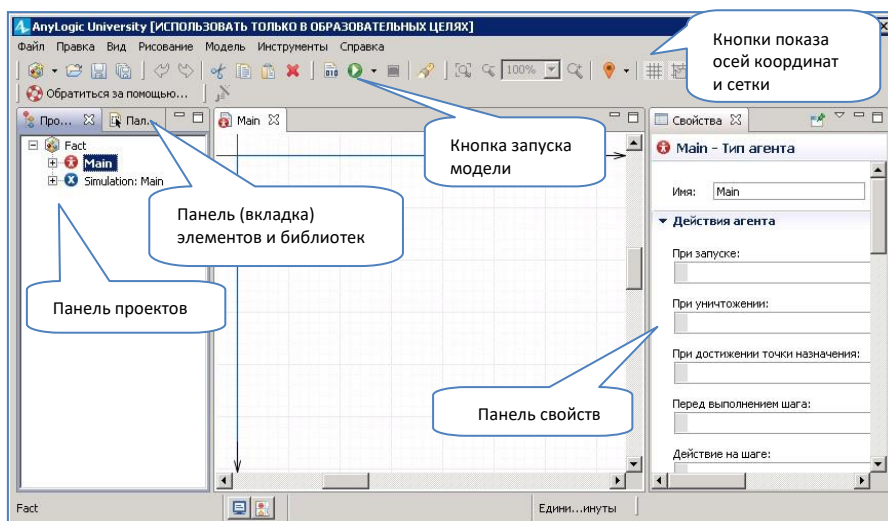


Рис. 2.2. Главное окно AnyLogic

Вернуть на прежнее место закрытое окно можно, активизировав соответствующий пункт главного меню *Вид* (рис. 2.3).

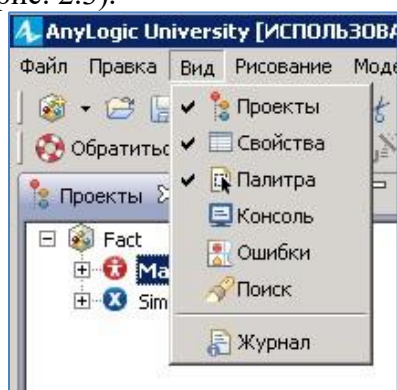


Рис. 2.3. Меню выбора рабочих окон

В современных средах программирования разработка приложений организуется в виде проектов.

В панели проектов AnyLogic отображаются древовидные диаграммы вновь созданных и загруженных с диска моделей. Если перед окончанием сеанса работы с программой они не были закрыты специальной командой меню *Файл–Закреть*, то при очередном запуске AnyLogic он загрузит их в рабочую среду и покажет в окне проектов. Эта опция весьма полезна, так как позволяет легко переносить (копировать) из одной модели в другую элементы, фрагменты кода и готовые классы.

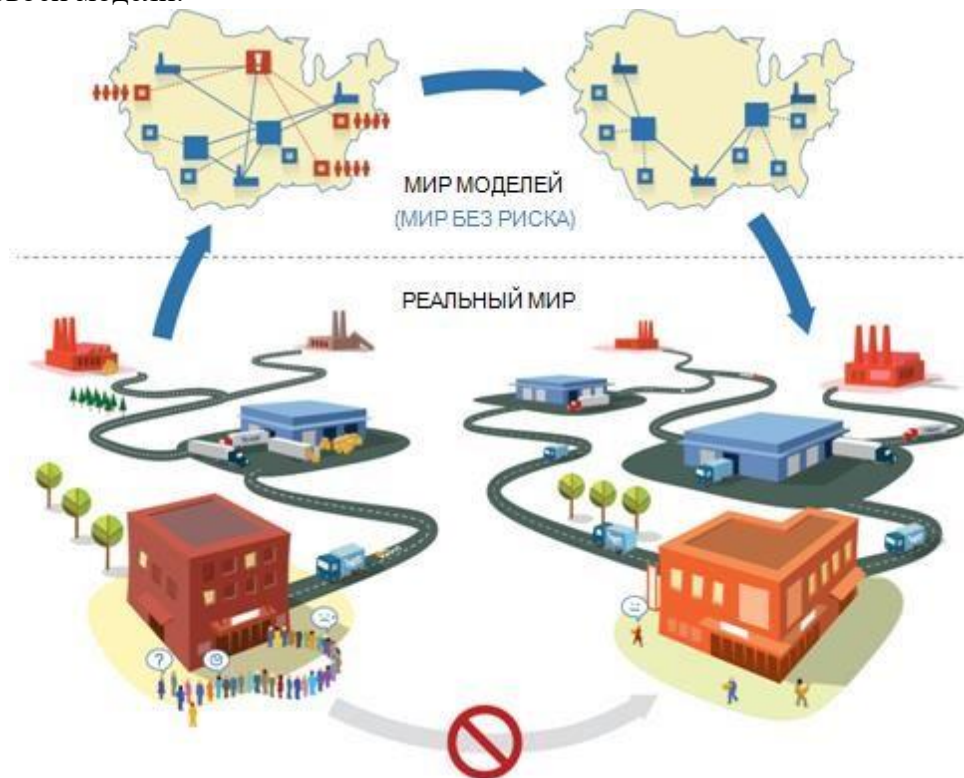
В начале работы с моделью дерево проекта содержит, как минимум, два компонента: класс активного объекта *Main* (главный класс, агент верхнего уровня) и простой эксперимент *Simulation:Main*.

Модель запускается в виде имитационного эксперимента кнопкой на панели инструментов или клавишей *F5*. Процесс отображается в специальном окне презентации. На начальном этапе проект представляет собой «пустую» заготовку, поэтому интерфейс и детали настроек эксперимента и окна презентации рассмотрим позже.

Создание модели в общем виде сводится к последовательному размещению в окне графического редактора функциональных элементов, определению их свойств, разработке новых классов и блоков программного кода и проведению экспериментов. Неизбежные спутники на этом пути – ошибки. Они отображаются в специальном окне с указанием места или объекта и достаточно подробным описанием проблемы на русском языке. Полезной будет подсказка кода, вызываемая комбинацией клавиш *CtrlSpace*, и отладчик, доступный в профессиональной версии AnyLogic.

2. Имитационное моделирование

Моделирование является одним из способов решения практических задач. Зачастую решение проблемы нельзя найти путем проведения натурных экспериментов: строить новые объекты, разрушать или вносить изменения в уже имеющуюся инфраструктуру может быть слишком дорого, опасно или просто невозможно. В таких случаях мы строим *модель* реальной системы, то есть описываем ее на языке моделирования. Данный процесс подразумевает переход на определенный уровень абстракции: опуская несущественные детали, мы учитываем только то, что считаем важным. Система в реальном мире всегда сложнее своей модели.



Моделирование

- ◆ **Все этапы разработки модели – проекция реального мира в мир моделей, выбор уровня абстракции и выбор языка моделирования – менее стандартизированы, чем процесс использования моделей для решения задач. Моделирование до сих пор больше искусство, чем наука.**

После создания модели – а иногда и в процессе разработки – мы начинаем исследовать структуру и понимать поведение системы, проверять, как она ведет себя при определенных условиях, сравнивать различные сценарии и оптимизировать ее. Когда оптимальное решение будет найдено, мы сможем применить его в реальном мире.

В сущности, моделирование является поиском решения задачи в защищенном от риска мире моделей, в котором мы можем ошибаться, отменять операции, возвращаться в прошлое и начинать все сначала.

Типы моделей

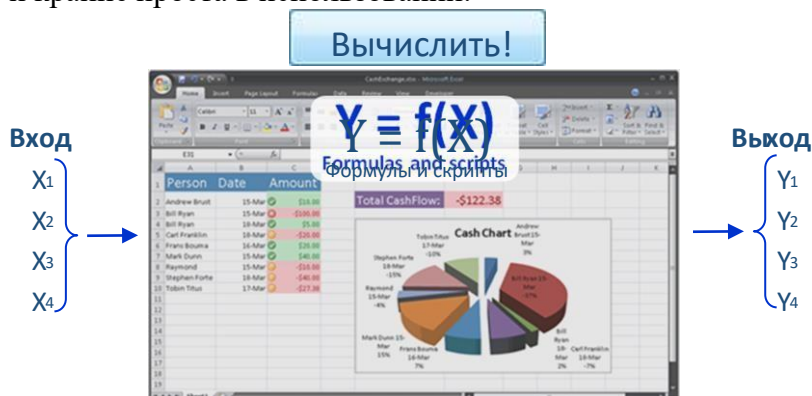
Есть много типов моделей, включая ментальные модели, которыми все мы пользуемся, чтобы понять, как устроен мир вокруг нас: друзья, семья, коллеги, город, в котором мы живем. Все наши решения - что следует сказать своему ребенку, что съесть на завтрак, за кого голосовать и в какой ресторан сходить на выходных, – основаны на ментальных моделях.

Мощным инструментом моделирования являются компьютеры, ведь они предоставляют легко управляемый виртуальный мир, в котором мы можем создать практически все, что способны представить. Конечно, существует множество различных типов компьютерных моделей: от электронных таблиц, позволяющих моделировать расходы, до сложных

инструментов имитационного моделирования, которые помогают исследовать динамические системы, например, потребительский рынок или зону боевых действий.

Сравнение аналитического и имитационного моделирования

Если вы спросите у сотрудников отдела стратегического планирования глобальной корпорации (равно как и у сотрудников отделов прогнозирования продаж, логистики, маркетинга, управления проектами и т.д.), какой инструмент моделирования они предпочитают использовать в своей работе, то самым популярным ответом будет: «Excel». У программы Microsoft Excel есть неоспоримые преимущества: она широко распространена и крайне проста в использовании.



Аналитическая модель (таблица MS Excel)

Суть технологии моделирования с использованием электронных таблиц крайне проста: вы вводите данные модели в одни ячейки и получаете выходные данные в других. Входные и выходные данные связаны формулами. Для задания дополнительной логики вы можете добавить в таблицу макросы. Различные надстройки позволяют вам выполнять эксперименты варьирования параметров, оптимизации или Монте-Карло.

Однако существует множество задач, для которых аналитическое (основанное на формулах) решение крайне сложно найти, а иногда оно и вовсе отсутствует. К таким задачам относятся в том числе и *динамические системы*, которым свойственно:

- Нелинейное поведение
- "Память"
- Неочевидные зависимости между переменными
- Причинно-следственные связи
- Неопределенность и большое количество параметров.

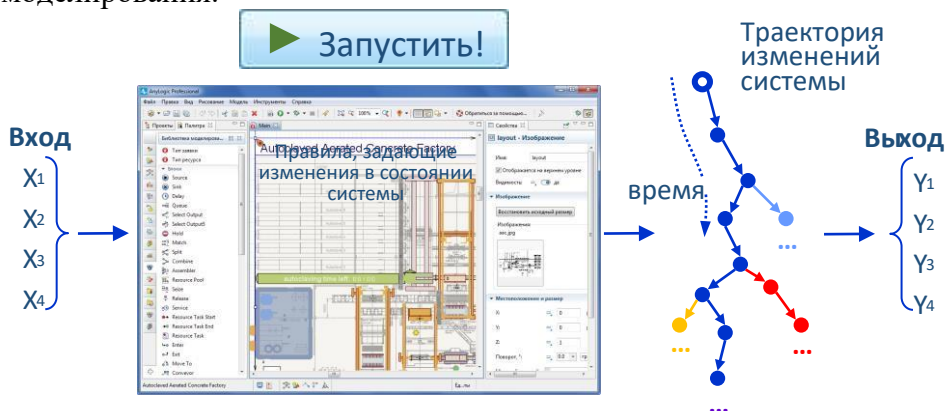
В большинстве случаев практически невозможно найти точные формулы и тем более построить ментальную модель такой системы.

Давайте рассмотрим задачу оптимизации автопарка логистической компании. Для ее решения необходимо учесть такие факторы, как расписание поездок, время погрузки и разгрузки товара, ограничения на время доставки и вместимость терминалов. При этом доступность транспортного средства в определенном месте в определенное время зависит от цепочки предшествующих событий, а выбор нового пункта назначения для свободной машины требует анализа связанных между собой будущих событий. Решить такую задачу средствами MS Excel практически невозможно.

□ Формулы, хорошо описывающие статические зависимости между переменными, как правило, плохо подходят для систем с динамическим поведением. Поэтому для анализа динамических систем мы используем другую технологию – имитационное моделирование.

Имитационную модель всегда является *выполняемой моделью*: вы запускаете ее, и она строит для вас траекторию изменений состояния системы. Можно сказать, что имитационная модель – это набор правил, согласно которым система переходит из одного состояния в другое. Правила могут задаваться самыми различными способами, например, дифференциальными уравнениями, диаграммами состояний, диаграммами процессов,

расписаниями. Выходные данные модели всегда можно проанализировать прямо по ходу моделирования.



Имитационная модель

Имитационные модели разрабатываются с помощью специализированного программного обеспечения, в котором используются различные языки моделирования. Для овладения навыком моделирования вам потребуется обучение, однако затраченное время и усилия окупятся, когда ваша модель предоставит высококачественный анализ сложной динамической системы.

Многие опытные пользователи MS Excel, владеющие навыками программирования, пытаются моделировать динамические системы с помощью электронных таблиц. В попытках учесть все больше и больше деталей, они неизбежно начинают воспроизводить функционал имитационных инструментов средствами в MS Excel. Получившиеся в результате модели работают очень медленно, в них невозможно разобраться, и вскоре выкидываются за ненадобностью.

Практически все перечисленные выше особенности динамических систем невозможно отразить в аналитической модели. Даже если общую конфигурацию системы удалось описать формулами, ее малейшее изменение может сделать эти формулы неверными, и для их исправления вам понадобится помощь профессионального математика.

Преимущества имитационного моделирования

Можно выделить шесть основных преимуществ имитационного моделирования:

1. Имитационные модели позволяют анализировать системы и находить решения в тех случаях, когда такие методы, как аналитические вычисления и линейное программирование не справляются с задачей.
2. После того, как вы определитесь с уровнем абстракции, разрабатывать имитационную модель будет гораздо проще, чем аналитическую, поскольку процесс создания модели будет инкрементальным и модульным.
3. Структура имитационной модели естественным образом отображает структуру моделируемой системы.
4. Имитационная модель позволяет вам отслеживать все объекты системы, учтенные в выбранном уровне абстракции, добавлять метрики и проводить статистический анализ.

Одним из главных преимуществ имитационного моделирования является возможность проигрывать модель во времени и анимировать ее поведение. Анимация будет неоспоримым преимуществом при демонстрации модели и может оказаться полезной для верификации модели и нахождения ошибок. Имитационные модели намного убедительнее электронных таблиц. Если вы используете имитационное моделирование, то при презентации проекта у вас будет явное преимущество перед теми, у кого на руках только цифры и решение, полученное из «черного ящика».

Области применения имитационного моделирования

Имитационное моделирование доказало свою успешность во многих областях применения. Появление новых методов моделирования и рост вычислительной мощности компьютеров позволяет утверждать, что количество этих областей будет только расти.



Рис. Области применения имитационного моделирования

На рисунке выше вы можете видеть распределение областей применения имитационного моделирования соответственно используемым в моделях уровням абстракции.

В нижней части рисунка располагаются модели физического уровня, в которых объекты реального мира моделируются максимально подробно. На этом уровне мы учитываем физическое взаимодействие, размеры, скорости, расстояния. Антиблокировочная система тормозов автомобиля, эвакуация болельщиков со стадиона, движение на регулируемом перекрестке, взаимодействие солдат на поле боя – все эти примеры требуют низкого уровня абстракции при их моделировании.

Модели, расположенные в верхней части схемы, более абстрактны и чаще всего оперируют обобщенными понятиями, такими как совокупность потребителей или статистика уровня занятости, а не отдельными объектами. Так как взаимодействие между объектами происходит на высоком уровне, такие модели помогают понять взаимосвязи в системе без необходимости моделировать промежуточные шаги, например, изучить влияние вложений в рекламу на продажи продукта компании.

Другие модели имеют средний уровень абстракции. Например, при моделировании отделения скорой помощи необходимо учитывать реальные размеры помещения, чтобы узнать, как долго пациент будет идти от приемной до рентгеновского кабинета. При этом, предположив, что помещение не переполнено, мы можем исключить из рассмотрения физическое взаимодействие между людьми.

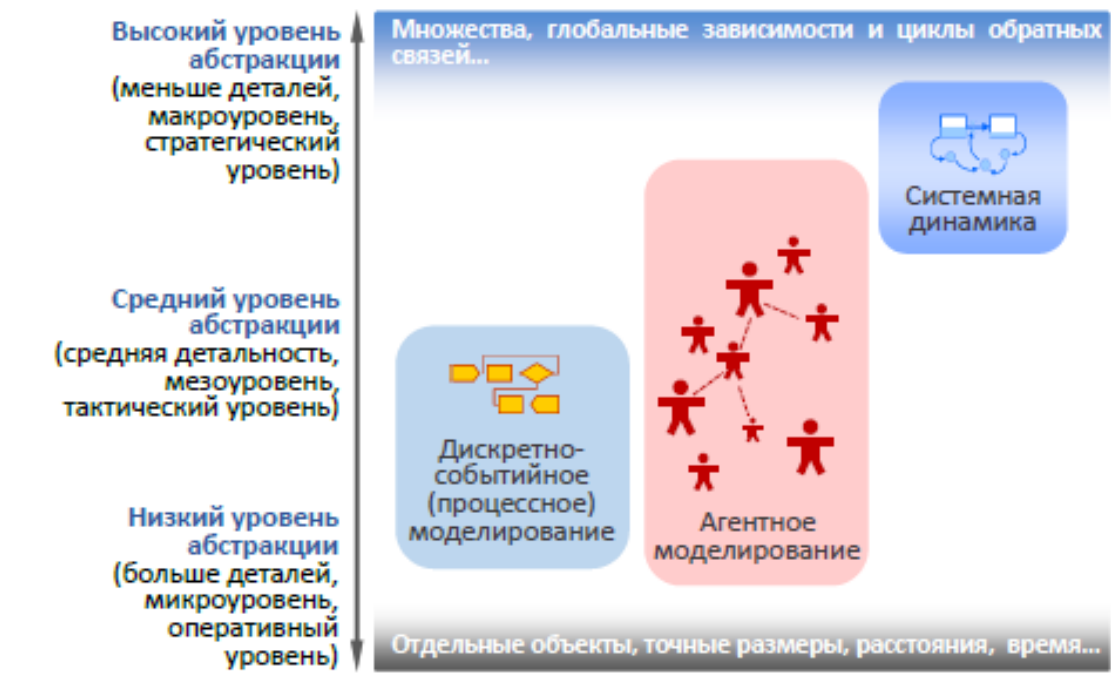
Разрабатывая модель бизнес-процесса или работы центра обработки звонков, мы моделируем последовательность и длительность операций, а не место, в котором они происходят. В модели грузоперевозок мы учитываем скорость грузовика или поезда, но в модели цепочки поставок на более высоком уровне мы просто считаем, что доставка заказа занимает от семи до десяти дней.

- ♦ **От правильности выбора уровня абстракции зависит успешность проекта моделирования. После того, как вы решите, что включать в модель, а что оставить за пределами уровня абстракции, выбрать метод моделирования будет уже не так сложно.**

- ♦ Нормально и даже ожидаемо, что в процессе разработки модели вам порой придется пересматривать выбранный уровень абстракции. В большинстве случаев вы начинаете с высокого уровня, а позже добавляете необходимые детали.

Три метода имитационного моделирования

В современном имитационном моделировании используются три подхода (методологии): дискретно-событийное моделирование, агентное моделирование и системная динамика.



Методы имитационного моделирования

В имитационном моделировании под *методом* понимается некая основа, которую мы используем, чтобы «перевести» систему из реального мира в мир моделей. Метод предполагает определенный язык, "положения и условия" для разработки модели. На данный момент, существует три метода:

- *Системная динамика*
- *Дискретно-событийное моделирование*
- *Агентное моделирование*

Каждый метод применяется в некотором диапазоне уровней абстракции. Системная динамика предполагает очень высокий уровень абстракции и, как правило, используется для стратегического моделирования. Дискретно-событийное моделирование поддерживает средний и низкий уровни абстракции. Между ними находятся агентные модели, которые могут быть как очень детализированными, когда агенты представляют физические объекты, так и предельно абстрактными, когда с помощью агентов моделируются конкурирующие компании или правительства государств.

Прежде чем выбрать метод моделирования, следует тщательно исследовать моделируемую систему и цели моделирования. На схеме ниже показано, что конкретная задача, стоящая перед разработчиком, во многом определяет подход к моделированию супермаркета. Разработчик может построить диаграмму процессов, в которой участвуют покупатели-заявки и кассиры-ресурсы, или агентную модель, в которой на покупателей-агентов влияет реклама и общение между собой и с сотрудниками-агентами компании, или диаграмму потоков и накопителей, в которой продажи связаны с рекламой, качеством сервиса, ценами и лояльностью клиентов.

Иногда отдельные части системы проще моделировать с помощью методов, отличных от основного. В таких ситуациях лучше всего строить многоподходные модели.

3. Агентное моделирование

Агентное моделирование - относительно новый метод моделирования. Поначалу оно являлось преимущественно предметом теоретических дискуссий в академических кругах, а начиная с 2000-х годов разработчики имитационных моделей стали использовать его на практике.

Переход к агентному моделированию был вызван:

- Желанием глубже изучить системы, которые сложно описать традиционными методами моделирования.
- Развитием технологии агентного моделирования (объектноориентированное моделирование, диаграммы состояний).
- Быстрому росту мощности процессоров и объема оперативной памяти компьютеров. Агентные модели более требовательны к ресурсам, чем модели системной динамики или дискретнособытийные модели.

Агентное моделирование предлагает разработчику моделей альтернативный взгляд на поведение системы.

□ **Вы можете не знать ни поведения системы в целом, ни ее главных переменных и зависимостей между ними, или не видеть четкой схемы процессов, но при этом понимать, как ведут себя отдельные элементы системы. В таком случае вы можете начать создание модели с идентификации моделируемых объектов (агентов) и задания их поведения. Иногда вам может понадобиться объединить агентов в сеть и позволить им взаимодействовать друг с другом, либо же поместить агентов в среду, которая имеет свою собственную динамику. Таким образом, глобальное поведение системы формируется из многих десятков (тысяч, миллионов) параллельно протекающих процессов.**

На данный момент не существует стандартного языка агентного моделирования. Структура агентной модели может быть задана как графически, так и с помощью сценариев. Поведение агента может быть задано различными способами. Если у агента есть состояние, от которого зависят его действия и реакции, то его поведение лучше всего задавать с помощью диаграммы состояний. Иногда поведение агента задается действиями, выполняемыми при наступлении определенных событий.

Иногда внутренняя динамика агента лучше всего задается с помощью дискретных событий или системной динамики. Так же и динамика среды, в которой живут агенты, может моделироваться с помощью традиционных методологий. По этой причине многие агентные модели совмещают в себе несколько подходов к моделированию.

Агентами могут быть самые разные объекты: транспортные средства, оборудование, проекты, организации, земельные участки, люди и так далее.

Ученые до сих пор спорят, какими именно свойствами должен обладать объект, чтобы называться агентом: способностью действовать и реагировать на действия других, ориентироваться в пространстве, обучаться, взаимодействовать и общаться, обладать «интеллектом» и т.п. На практике в агентных моделях вам могут встретиться агенты любых типов: одни общаются друг с другом, а другие находятся в полной изоляции; одни живут в пространстве, а другие – нет; одни обучаются и приспосабливаются, а другие никогда не меняют своего поведения.

Приведем несколько полезных фактов об агентах, чтобы многообразие теорий не вводило вас в заблуждение:

- **Агенты не являются клеточными автоматами** и не обязательно обитают в дискретном пространстве (как в игре «Жизнь»). Во многих агентных моделях пространство вообще отсутствует. Когда пространство все же необходимо, оно чаще всего является непрерывным (это может быть карта мира или план здания).

Агенты – не обязательно люди. Агентом может быть все, что угодно: транспортное средство, оборудование, проект, организация или даже идея.

Агентом может быть объект, кажущийся абсолютно пассивным. Например, в модели нефтепровода вы можете представить сегмент трубы как агента, задав для него графику техобслуживания, вероятности происхождения аварий, логику проведения ремонтных работ, затраты и т.д.

Агентов в модели может быть как много, так и мало. При этом агенты могут быть как одного типа, так и разных.

Существуют агентные модели, в которых агенты вообще не взаимодействуют друг с другом. Например, в моделях потребления алкоголя, развития ожирения или хронических заболеваний индивидуальная динамика агента зависит только от его личных параметров и, в некоторых случаях, от среды.