

**Донецкий национальный технический
университет**

**Методические указания
к лабораторным работам по курсу:
«Языки и системы имитационного
моделирования»**

**для студентов специальности
«Программная инженерия»**

2021

**Составитель: проф. каф. ПМИ, к.т.н.
Григорьев Александр Владимирович**

Оглавление.

Лабораторная работа 1. Выбор темы.

Лабораторная работа 2. Разработка вербальной модели для методики проектирования в заданной предметной области.

Лабораторная работа 3. Разработка средствами диаграмм IDEF0 статических моделей, имеющихся в вербальной модели для заданной предметной области.

Лабораторная работа 4. Разработка средствами диаграмм типа IDEF3 динамических моделей, имеющихся в вербальной модели для заданной предметной области.

Лабораторная работа 5. Разработка имитационной модели управления объектом средствами системы Arena для CALS-технологии заданной предметной области.

Лабораторная работа 6. Обеспечение адекватности имитационной модели управления объектом, построенной средствами системы Arena.

Лабораторная работа 7. Исследование имитационной модели управления объектом средствами системы Arena для CALS-технологии заданной предметной области.

Приложение 1. *Система имитационного моделирования ARENA 7.0. Basic Process Panel.*

Приложение 2. *Пример исследования имитационной модели управления объектом средствами системы Arena.*

Лабораторная работа 1. Выбор темы.

Необходимо сформулировать тему явно.

Например: Моделирование процесса функционирования гражданского самолета АН24.

Тема модели соответствует теме из курса «Анализ требований программного обеспечения».

Время – 2 часа.

Лаб. 2.

Разработка вербальной модели для методики проектирования в заданной предметной области.

1. Порядок формирования вербальной методики проектирования

Вербальные описания должны содержать информацию:

- о сущности и свойствах системы как единого целого (**входные и выходные данные системы, назначение системы**);
- о составе, свойствах частей и их функциях (**структура системы, входы-выходы частей, их задачи**);
- о структуре и ее роли в формировании свойств системы, как единого целого (**как предложенный порядок обеспечивает связь входов и выходов системы как единого целого**);
- о взаимосвязи и функциях подсистем (**какие части, в какие подсистемы входят, функции этих подсистем**);
- о возможных изменениях в составе и функциях системы (**альтернативные варианты структур, входов-выходов**);
- об экспериментальных данных, характеризующих структуру и поведение системы (**типичные структуры, ГОСТы, СПИПы и т.д.**).

Фактически это - полноценный текст, содержательно описывающий модель объекта.

Для задания вербальной методики проектирования необходимо сформировать в вербальном виде такие исходные документы:

- 1) Описание бизнес-процессов;

Рассмотрим их детальнее.

Описание бизнес-процессов

Модель создаваемой САПР в определенной мере должна отражать модель организации системы. Соответственно, другим важным источником информации, являются артефакты, описывающие предметную область. Это могут быть **документы с описанием бизнес-процессов предприятия**, выполненные консалтинговым агентством, либо просто документы (должностные инструкции, распоряжения, своды бизнес-правил), принятые на предприятии.

1) ОПИСАНИЕ БИЗНЕС-ПРОЦЕССА ПРЕДПРИЯТИЯ

Описание бизнес-процессов предприятия должно включать такие общие описания:

Модель объекта проектирования:

- 1) Статическую модель объекта проектирования;
- 2) Функциональную модель объекта проектирования;

Отчет:

1) Неформально т.е. словесно в свободном формате прописанные требования в соответствии с классификацией источников и стратегий выявления требований, включая:

- документы с описанием бизнес-процессов предприятия;

Лаб. 3

Разработка средствами диаграмм IDEF0 моделей бизнес-процессов, имеющих в вербальной модели для заданной предметной области.

В среде пакета BrWin (AllFusion Process Modeler) построить модели процессов:

1) Статическую модель объекта проектирования:

2) Статическую модель проектной организации.

Источник информации: вербальная модель.

Требования:

- не менее трех уровней (начальный блок, его структура т.е. подблоки, структура подблоков);

- не менее трех подблоков для каждого разбитого блока.

Отчет:

- все диаграммы.

Порядок создания моделей.

Нажав правую кнопку на начальном блоке в разделе «font» выбрать русский язык и применить его ко всем блокам.

Нажав правую кнопку и выбрав «name», задать имя блока.

Аналогично сделать для всех стрелочек (связей).

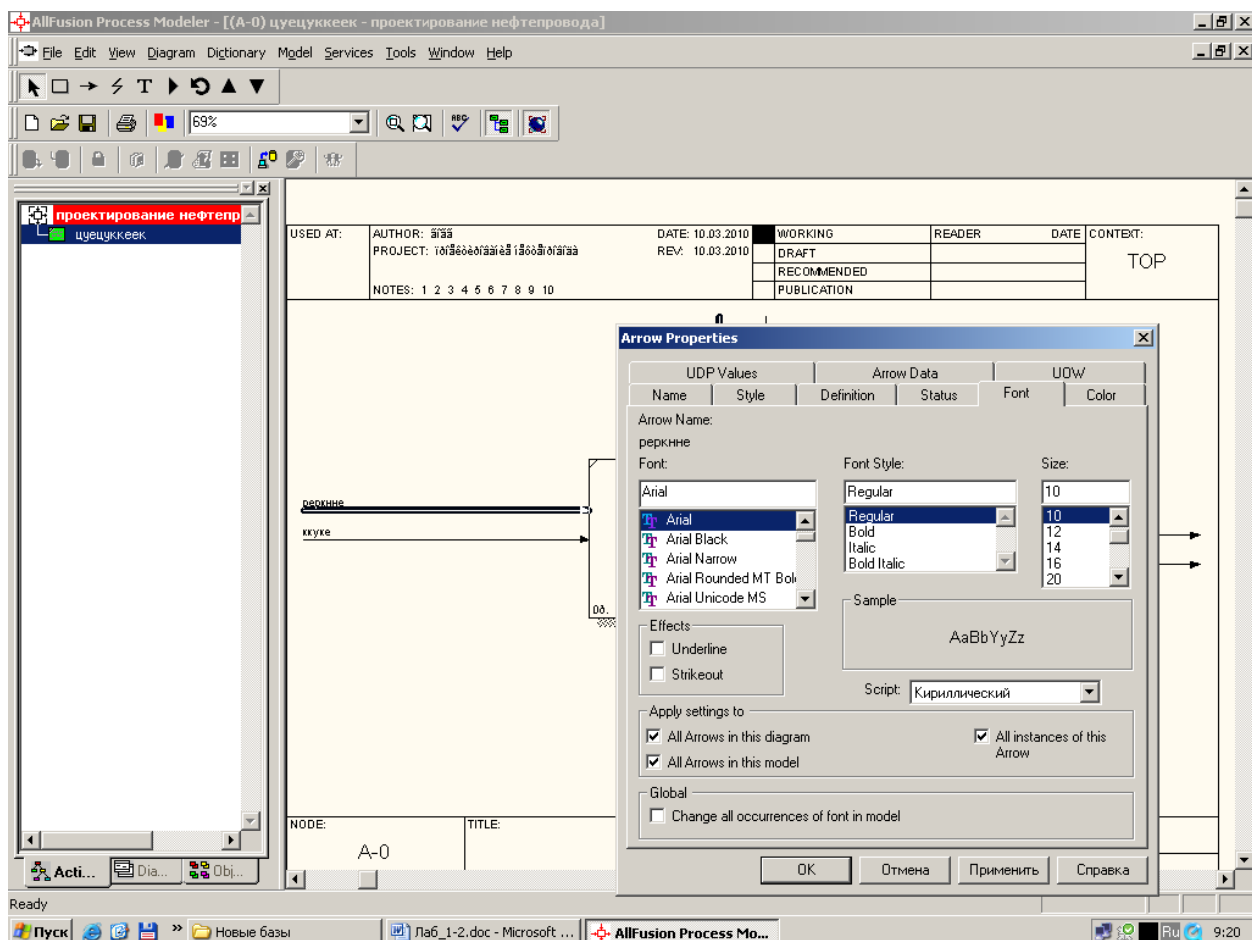


Рис. Пример установки русского языка для всех стрелочек.

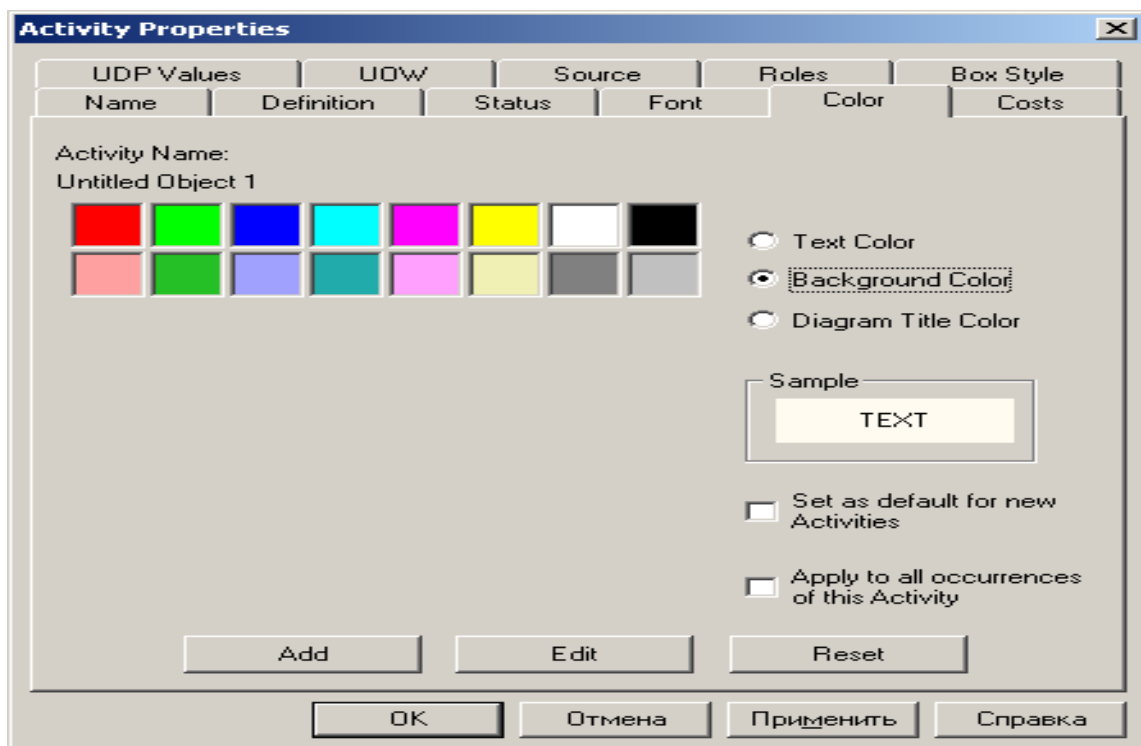


Рис. Меню раскраски блоков.

Лаб. 4.

Разработка алгоритмической модели управления объектом средствами диаграмм типа **IDEF3** для CALS-технологии заданной предметной области.

Пользуясь средствами диаграмм типа **IDEF3** описать алгоритмическую модель для заданной предметной области, включая:

- 1) Динамическую модель объекта проектирования;
- 2) Динамическую модель процесса проектирования:

Использовать:

- все типы блоков-перекрестков;
- не менее трех уровней иерархии, применяя на каждом уровне не менее трех блоков (перекрестки не учитываются).

Отчет по Лабе включает:

- 1) Не менее 5-ти диаграмм для управление процессом

Требование: Использовать тип **PFDD** диаграммы.

Срок: следующий раз.

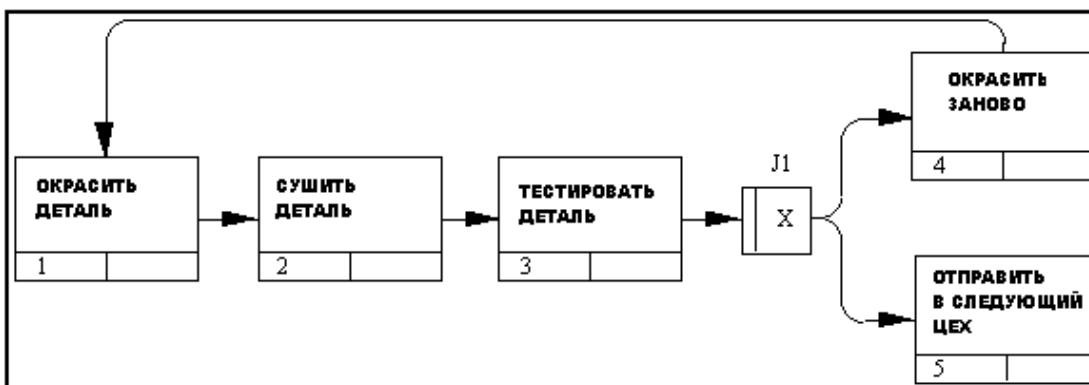


Рисунок 1. Пример PFDD диаграммы.




Пример IDEF3-диаграммы

Рисунок 1. Пример PFDD диаграммы.

Типы перекрестков:

	Наименование	Смысл в случае слияния стрелок	Смысл в случае разветвления стрелок
	Асинхронное И.	Все предшествующие процессы должны быть завершены	Все следующие процессы должны быть запущены
	Синхронное И.	Все предшествующие процессы завершены <i>одновременно</i>	Все следующие процессы запускаются <i>одновременно</i>
	Асинхронное ИЛИ.	Один или несколько предшествующих процессов должны быть завершены	Один или несколько следующих процессов должны быть запущены
	Синхронное ИЛИ.	Один или несколько предшествующих процессов завершаются	Один или несколько следующих процессов запускаются <i>одновременно</i>

		<i>одновременно</i>	
	Исключающее ИЛИ.	Только один предшествующий процесс завершен	Только один следующий процесс запускается

Лаб. 5.

Разработка имитационной модели управления объектом средствами системы Arena для CALS-технологии заданной предметной области.

Необходимо установить инсталляцию системы имитационного моделирования ARENA.

Затем, используя предварительно построенные диаграммы типа **IDEF3** вручную (или путем экспорта из меню файл системы AlfusionModeling или системы BpWin) построить аналогично построенную имитационную модель управления объектом, но выполненную средствами системы Arena для CALS-технологии заданной предметной области.

Должны рассматриваться алгоритмы:

- управление процессом функционирования объектом во внешней среде.

Использовать:

- все диаграммы, имеющие место в алгоритмической модели управления объектом, описанной средствами диаграмм типа **IDEF3**;

Отчет по Лабе 2 включает:

2) Диаграмму модели, построенную в системе Arena;

Срок: 5 недель.

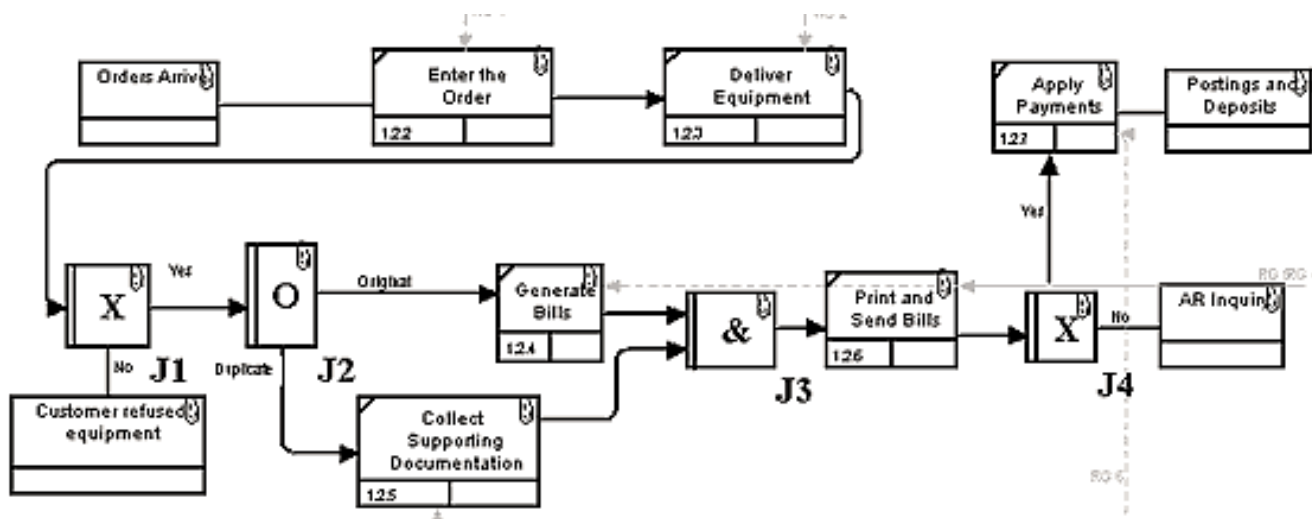


Рис. 1. Диаграмма IDEF3 – пример для иллюстрации экспорта в Arena.

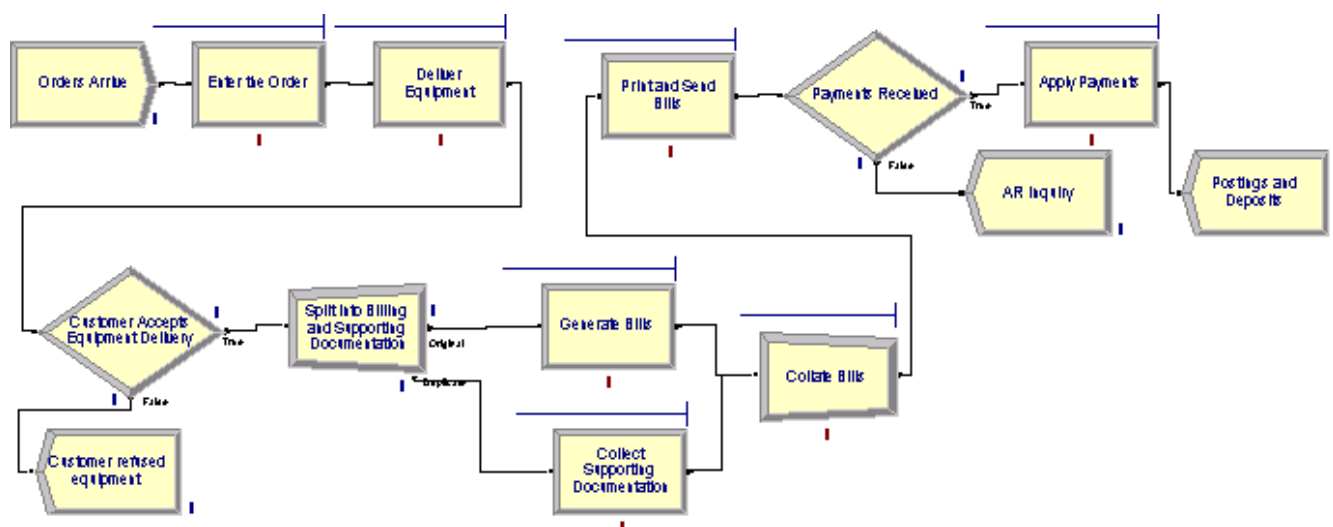


Рис. 2. Имитационная модель Arena – результат импорта из BPwin.

Порядок выполнения работы:

1) В соответствии с диаграммой типа IDEF3 построить вручную путем использования технологии «возьми и брось» соответствующую имитационную модель в системе Arena.

Основные требования:

1) В процессе создания модели использовать все графические типы блоков:

1. Источники заявок (Create);
2. Процессы (Process);
3. Блоки вероятностного изменения движения сообщений (Decide);
4. Блоки создания параллельных копий заявок (Separate);
5. Блоки сбора параллельных заявок в одну (Batch);
6. Уничтожители заявок (Dispose);
7. Блоки назначения параметров заявок и переменных (Assign);
8. Блоки задания статистики (Record).

2) Для задания в модели типов заявок, параметров заявок, наименований и размеров ресурсов и т.д. использовать все модули данных:

1. Модуль Entity (сущности)
2. Модуль Queue (очереди)
3. Модуль Resource (ресурсы)
4. Модуль Schedule (расписания, вспомогательная информация)
5. Модуль Set (определения)
6. Модуль Variable (переменные)

Дополнительные требования:

1) Использовать в блоках Process (процессы) все ресурсы, используемые ранее в диаграммах типа IDEF0;

- необходимо обеспечить занятие, задержку и освобождение всех ресурсов (см. пример);

2) В блоках Assign (назначения параметров):

- для параметров заявок использовать все базовые законы распределения случайных чисел (нормальный, треугольный, экспоненциальный);

- задать пиктограммы для всех типов заявок (см. пример);

3) В блоках условного ветвления (Decide) использовать все возможные способы ветвления;

4) Для отображения вышележащих структурных блоков, имевших место в IDEF3, использовать вспомогательные графические блоки типа «рамка» (см. пиктографическое меню Arena) с указанием текстового наименования вышележащего блока. Вариант реализации приведен в примере.

5) Для отображения результатов моделирования необходимо для наиболее важных показателей (3-4) вывести с помощью пиктографического меню соответствующие графические образы - диаграммы и т.д. (см. прилагаемый пример).

6) Необходимо использовать механизм «стоимость» для сбора дополнительной информации.

Полный набор информации по Arena на сайте:

<http://www.aics.ru/books.shtml?a=showbooks>

Пример выполнения работы в файле Model3_1.doc. Все, что реализовано в данном примере, обязательно для выполнения.

Лабораторная работа 6. Обеспечение адекватности имитационной модели управления объектом, построенной средствами системы Arena.

Необходимо обеспечить адекватность имитационной модели на уровне, при котором результаты экспериментов над моделью позволят принимать обоснованные решения по изменению структуры и алгоритма работы реального объекта.

Адекватность модели реальной системе определяется уровнями достоверности модели.

Табл.1 Уровни достоверности модели:

	Тип модели	Характеристика модели
1	КАЛИБРОВАННАЯ (проверенная модель)	Параметры моделей установлены в соответствии с имеющимися экспериментальными данными о реальной системе.
2	ОБОСНОВАННАЯ (верифицированная модель)	Поведение калиброванной модели схоже с поведением реальной системы на уровне ее входов-выходов.
3	РЕАЛИСТИЧНАЯ (валидная модель)	Поведение обоснованной модели схоже с поведением реальной системы на уровне ее элементов, внутренних связей, отдельных подпроцессов.

Уровень адекватности конструктивной модели должен быть предельно высоким, исходя из возможностей привлеченной математической теории.

Модель, построенная в лаб. 2, имеет параметры всех блоков (процессы, ветвления и т.д.), установленные в соответствии с реальными представления о системе. Конкретно это касается:

- размеров ресурсов;
- параметров ожидания в очереди (например, мат. ожидание для нормального закона, задающего время ожидания в очереди и т.д.);
- вероятности условных переходов по различным путям и т.д.

Т.о., данная модель соответствует уровню 1, т.е. – калиброванной модели, при которой исходные параметры модели установлены в соответствии с данными о реальной системе.

Необходимо обеспечить дополнительно уровни 2 и 3.

1) Обеспечение уровня верифицированной модели

Для обеспечения уровня адекватности 2, т.е. – верифицированной модели, необходимо задать данные о реальном эксперименте на системе на уровне входов-выходов системы.

Определить входы системы означает:

- задать интенсивности поступления всех типов входных сущностей (транзактов);
- задать параметры всех типов входных сущностей.

Определить выходы системы означает иметь реальную статистику на выходе модели для:

- всех типов выходных сущностей;
- параметров всех выходных сущностей.

Например, нам известно следующее:

1) Реальный эксперимент длился 2 часа;

2) За это время на входе системы:

- вошло 40 сущностей «куры»;
- вошло 60 сущностей «гуси»;
- общий вес «кур» составил 1 т;
- общий вес «гусей» составил 1,4 т;

2) За это время на выходе системы:

- вышло 20 сущностей «куры»;
- вышло 40 сущностей «гуси»;
- отбраковано 5 «кур» и 3 «гуся»;
- среднее время, которое потратил «гусь» на выходе системы, составила 20 мин., минимальное – 10 мин., максимальное – 20 мин.

мин., минимальное – 10 мин., максимальное – 20 мин.

- среднее время, которое потратила «курица» на выходе системы, составила 25 мин., минимальное – 15 мин., максимальное – 25 мин.

- общий вес «кур» составил 0,5 т;

- общий вес «гусей» составил 1 т;

Вначале устанавливаем входные параметры модели так, что бы обеспечить требуемую статистику для входов системы.

Цель последующего эксперимента на модели – исследование соответствия выходной модельной статистики заданной реальной статистике.

Если соответствия нет, то изменяем внутренние параметры модели так, что бы обеспечить требуемую статистику для выходов системы. Например:

- меняем среднее время (мат. ожидание) задержки операции «жарить»;
- и т.д.

Порядок выполнения работы:

- 2) Задать статистику реального эксперимента на уровне входов-выходов;
- 3) Исследовать модель и сформировать выходную модельную статистику;
- 4) Если модельная статистика не соответствует реальной, то – сделать изменения модели и обеспечить требуемую статистику.
- 5) Если задача выполнена, то необходимый второй уровень адекватности модели достигнут;
- 6) Если нет, то возвращаемся на пункт 2.

Отчет:

- реальная статистика;
- модельная статистика (некорректная);
- предлагаемые изменения в модели;
- модельная статистика (корректная);

2) Обеспечение уровня валидной модели

Для обеспечения уровни адекватности 3, т.е. – валидной модели, необходимо задать данные о реальном эксперименте, описывающей поведение системы уже для отдельных подпроцессов в верифицированной модели.

Конкретно это означает задать реальную статистику о:

- среднем, минимальном и максимальном размере очереди на всех блоках;

- среднем, минимальном и максимальном времени ожидания всех типов сущностей на всех блоках;

- среднем, минимальном и максимальном проценте использования (утилизации) всех типов ресурсов на всех блоках;

Например:

- 1) Реальный эксперимент длился 2 часа;

- 2) За это время:

- процент утилизации ресурса «сковорода» в блоке «жарить» составил 40;

- средняя длина очереди к блоку «жарить» составила 1,5 сущностей, минимальное – 1, максимальное -3;

- средняя время ожидания в очереди к блоку «жарить» составила 7 мин., минимальное – 3, максимальное -10;

- и т.д.

Цель последующего эксперимента на модели – исследовать соответствия выходной модельной статистики заданной реальной статистике.

Порядок выполнения работы:

- 1) Использовать верифицированную модель

- 2) Задать статистику реального эксперимента на уровне подпроцессов;

- 3) Исследовать модель и сформировать внутреннюю модельную статистику;

- 4) Если модельная статистика не соответствует реальной, то – сделать изменения модели и обеспечить требуемую статистику.

- 5) Если задача выполнена, то необходимый третий уровень адекватности модели достигнут;

- 6) Если нет, то возвращаемся на пункт 2.

Отчет:

- реальная статистика;

- модельная статистика (некорректная);

- предлагаемые изменения в модели;

- модельная статистика (корректная).

Время – 2 часа.

Срок: 1 неделя.

Лабораторная работа 7. Исследование имитационной модели управления объектом средствами системы Arena для CALS-технологии заданной предметной области.

1. Задача планирования машинных экспериментов, основные понятия и определения

При планировании экспериментов мы рассматриваем объект как «чёрный ящик». Цель эксперимента – нахождение зависимостей между выходными характеристиками объекта и некоторыми входными величинами.

Выходная величина y – *эндогенная*, если мы способны её измерить. Входные параметры X_1, X_2, \dots, X_n – факторы. Функция $y = f(X_1, X_2, \dots, X_n)$ называется функцией или поверхностью отклика. Данная функция показывает зависимость выходной эндогенной величины от факторов.

Поскольку вид функции заранее неизвестен, то фактически решается задача регрессии. В этом случае чаще всего в качестве функции отклика выбирают некоторый отрезок – ряд Тейлора, например

$$y = \beta_0 + \sum_{i=1}^n \beta_i \cdot x_i + \sum_{i,j=0}^n \beta_{ij} \cdot x_i \cdot x_j + \sum_{i=1}^n \beta_{ii} \cdot x_i + \dots$$

По существу, по результатам эксперимента мы отыскиваем коэффициенты данного ряда, т.е. β_i, β_{ij} и т.д.

Для факторов, которые мы можем изменять при исследовании модели, существуют некоторые интервалы возможных значений. Исходя из этого, рассмотрим величины:

x_i^0 – основные (начальные) значения факторов;

x_i^{\max} – верхний уровень значимости фактора;

x_i^{\min} – нижний уровень значимости фактора;

Δx_i – интервал изменения.

Начальное значение фактора – это предыдущее значение фактора, которое используется для выбора следующего значения фактора.

Если рассматривать урезанный подход в проведении эксперимента, рассматривается только \min и \max значения факторов (*экстремальный эксперимент*), тогда возможно вести кодировку:

«+1» – \max уровень;

«- 1» – \min уровень.

При таком двухуровневом способе задания значений факторов число всех точек, для которых необходимо провести эксперимент, составляет 2^n , где n – число факторов.

2. Факторные планы

В случае эксперимента над моделью число экспериментов составляет $N = 2^n$. Использование всех возможных комбинаций граничных значений факторов называется полным факторным планом. Например, для $n = 3$:

Табл. 1.

Номер пробы	$\overline{x_1}$	$\overline{x_2}$	$\overline{x_3}$	y
1	-1	-1	+1	y1
2	+1	-1	-1	y2
3	-1	+1	-1	y3
4	+1	+1	+1	y4
5	-1	-1	-1	y5
6	+1	-1	+1	y6
7	-1	+1	+1	y7
8	+1	+1	-1	y8

Полные факторные планы имеют следующие важные для планирования эксперимента свойства:

1) Симметричность плана относительно центра эксперимента

$$\sum_{j=1}^N \overline{x_{ij}} = 0, \quad (i = 1, 2, \dots, n),$$

где:

$\overline{x_{ij}}$ - значение i-го фактора в j-м эксперименте.

2) Нормирование плана

$$\sum_{j=1}^N \overline{x_{ij}^2} = N, \quad (i = 1, 2, \dots, n)$$

3) Ортогональность, т.е. скалярное произведение вектор-столбцов матрицы плана равно нулю

$$\sum_{j=1}^N \overline{x_{ij}} \cdot \overline{x_{uj}} = 0, \quad (i \neq u; \quad i, u = 1, 2, \dots, n)$$

4) Рототабельность

Здесь «roto» – вращаемость. Означает, что точность прогнозного значения функции отклика является одинаковой на различных расстояниях от центра эксперимента и не зависит от направления движения. Расстояние – это норма вектора в многомерном линейном пространстве.

Полные факторные планы позволяют в принципе определить не только меру влияния каждого отдельного фактора на поведение системы, но также выявить влияние отдельных комбинаций значений факторов, т.е. имеется возможность проследить т.н. эффект взаимодействия.

В этом случае в матрицу эксперимента дополнительно вводятся столбики, соединяющие возможные комбинации значений X, например $X_1 * X_2$; $X_2 * X_3$.

Соответственно, для них записываются значения +1 или -1, что позволяет в дальнейшем при регрессии явно отследить зависимости выхода от искомых комбинаций факторов.

Полные факторные планы являются наиболее корректным способом исследования моделей, однако на практике в связи с большой трудоемкостью таких экспериментов используются дробные факторные планы.

В этом случае могут рассматриваться не все возможные комбинации значений факторов, а некоторые их подмножества. Однако требования, предъявляемые к другим факторным планам, совпадают с требованиями к

полным факторным планам, т.е. симметричность, нормированность, ортогональность, ротатабельность.

Чаще всего на практике применяются факторные планы, составляющие половину исходного полного факторного плана. Такие планы называются полурепликами, в отличие от полных планов, называемых репликами.

Имеются четверть-реплики и т.д.

3. Пример создания факторного плана для экспериментов над ИМ в системе Arena.

В рамках единичного запуска модели на достаточно длительном периоде моделирования мы сможем узнать следующие данные об объекте:

- среднюю длину очереди в некотором процессе;
- средний процент утилизации ресурсов;
- среднюю стоимость одной сущности и т.д.

Это позволит нам сделать вывод о качестве объекта моделирования и о его узких местах.

Т.о., можно внести исправления в модель с целью ее улучшения.

Однако, недостатком такого подхода является:

- наличие исследования модели только в одном выбранном режиме работы объекта (интенсивности заявок с данным конкретным мат. ожиданием и дисперсией, один из режимов работы оборудования, заданного как задержки на обслуживание и т.п.);

- неизвестность относительно прочих возможных режимов работы объекта и его качестве в этих случаях.

Т.о., мы не можем сделать качественный вывод о качестве системе в целом.

Следовательно, необходим факторный анализ модели.

После создания таблицы факторного анализа можно построить уравнение поверхности отклика «у» как функцию (например, полином некоторой степени) от переменных, в данном случае (см. таб. 1):

$$y=f(\bar{x}_1, \bar{x}_2, \bar{x}_3).$$

Построить такую функцию можно на полном факторном плане, или – на полуреплике и т.д.

Как правило, поверхность отклика достигает своего экстремума в одной из комбинаций параметров полного факторного плана.

Но для поиска точного значения экстремума поверхности отклика можно выполнить поиск экстремума на всей поверхности отклика, заданной своим уравнением. Например, методом ветвей и границ.

Пусть, к примеру, «у» - это такой важный параметр модели, как утилизация (процент использования) ресурса «сковороды».

Параметрами могут быть:

\bar{x}_1 - мат. ожидание нормального закона генерации сущностей «гусь»;

\bar{x}_2 - мат. ожидание закона генерации сущностей «курица»;

—
 x_3 - вероятность появления несвежей птицы.

Выбрав мин. и макс. для каждой переменной, построим полный факторный план, аналогичный табл. 1.

На его основе построим полуреплику и проделаем соответствующие эксперименты (4 шт.).

Построим поверхность отклика утилизации как регрессию полиномом 2-го порядка.

Отыщем на данном полиноме, например в EXCEL, мин и макс.

Это уже будут более достоверные, более полные данные о модели.

Сделаем выводы о качестве объекта по полученным параметрам модели.

Например:

1) При любых возможных комбинациях параметров модели утилизация ресурса «сковорода» равна 1, что говорит о не оптимальности исследуемого технологического процесса. После этого можно предложить изменить структуру объекта, а затем, проделав подобные эксперименты, исследовать новую модель и т.д.

2) При некоторых возможных комбинациях параметров модели утилизация ресурса «сковорода» равна 1, что говорит о не оптимальности исследуемого технологического процесса. После этого можно предложить ограничения на режимы работы объекта (интенсивность «гусей» не более K , ...).

4. Отчетность

Отчет по Лабе 7 включает:

- 3) Определение эндогенных (выходных) параметров (1);
- 4) Определение для каждого эндогенного параметра 3 входных параметров (факторов);
- 5) Определение для них значений мин. и макс.
- 6) Создание полного факторного плана;
- 7) Исследование по каждому факторному плану модели с заданием соответствующих входных значений мин. или макс.
- 8) Получение значения эндогенной переменной;
- 9) Регрессия для результатов факторного плана;
- 10) Отрисовка поверхности отклика;
- 11) Поиск экстремумов (мин. и макс.) для регрессионной зависимости;
- 12) Корректировка модели, исходя из полученных экстремумов;
- 13) Повторное единичное исследование модели;
- 14) Оценка эффективности сделанных изменений.

Срок: 2 недели.

Приложение 1.

Система имитационного моделирования ARENA 7.0. Basic Process Panel

Методические указания к выполнению лабораторных работ по курсам: «Моделирование систем» для студентов специальности 220400 «Программное обеспечение вычислительной техники и автоматизированных систем».

Введение

Система имитационного моделирования Arena – основной программный продукт корпорации Systems Modeling. В основе ПП Arena лежит язык SIMAN.

Arena позволяет создавать динамические модели разнородных процессов и систем в терминах предметной области, оптимизировать построенную модель.

Программный продукт Arena 7.0 позволяет создавать стохастические динамические модели любого предприятия. Динамические модели позволяют оптимизировать, прогнозировать, планировать деятельность предприятий, а также проводить анализ деятельности предприятия на основании полученных моделей и выдавать рекомендации по улучшению работы конкретного предприятия.

В результате моделирования Arena 7.0 формирует отчеты, в которых можно посмотреть отчеты по:

- объектам, находящимся в системе (общее время нахождения в системе, суммарное время ожидания объекта в системе, количество объектов, вошедших/вышедших в систему/из системы);
- очередям, образующихся в модулях процессов, если ресурс захвачен другим объектом (время ожидания обработки в очереди, количество объектов, ожидающих в очереди);
- процессам – статистика для каждого повторения;
- ресурсам – статистика по затраченным ресурсам; пользователь сам определяет статистику, которая будет отражена в данном отчете.

По результатам собранной статистики если не устраивают какие-либо параметры, то изменение свойств модулей приведет к изменению всех параметров. Т.е. методом прогона различных вариаций параметров модели можно определить оптимальный вариант работы созданной системы. В этом и есть плюс, т.к. шлифовка работы предприятия делается сначала на модели и только затем внедряется на производство.

Также существует возможность экспорта статических моделей BPWin в динамические модели Arena.

1. Начало работы с Arena 7.0

1.1. Создание модели с помощью ПП Arena 7.0

Для того чтобы создать новую модель необходимо открыть ПП Arena 7.0 через Пуск; Rockwell Software; Arena7.0; Arena7.0. После запуска Arena автоматически открывается новый файл. Модули помещаются на панель методом «drag & drop», соединяются с помощью коннектора. Если модуль остается «горячим» (т.е. выделенным), то при помещении нового модуля на рабочую область (окно блок-схемы) эти модули автоматически соединяются друг с другом.

1.2. Интерфейс ПП Arena 7.0

Среда моделирования Arena представлена на рисунке 1.

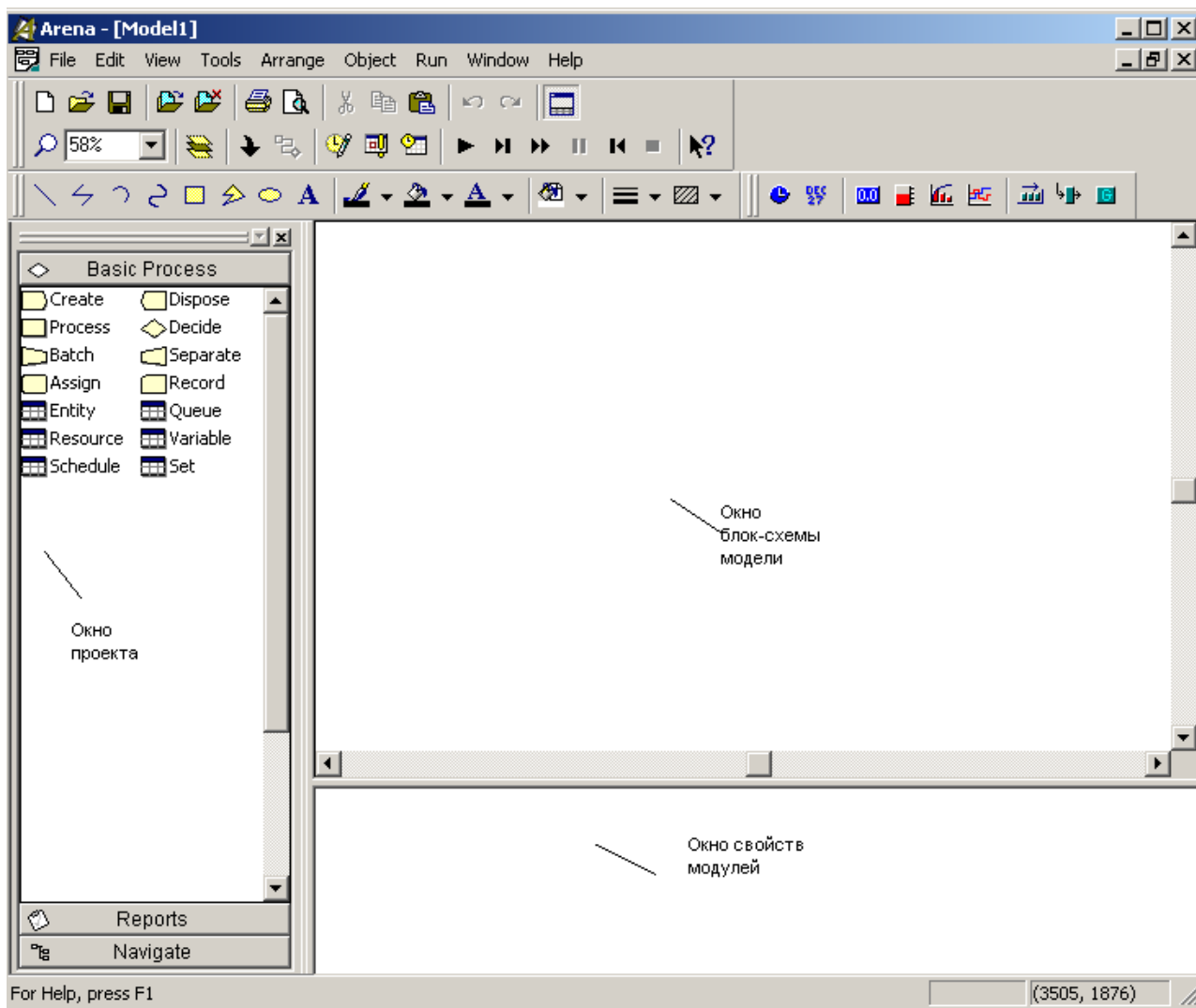


Рисунок 1 – Среда моделирования Arena

Окно приложения разделено на три области:

- Окно рабочего поля
- Окно свойств модулей
- Окно проекта

Окно проекта включает в себя несколько панелей:

- **Basic Process** (панель основных процессов) – содержит модули, которые используются для моделирования.
- **Reports** (панель отчетов) – панель сообщений: содержит сообщения, которые отображают результаты имитационного моделирования.
- **Navigate** (панель навигации) – панель управления позволяет отображать все виды модели, включая управление через иерархические подмодели.

Окно рабочего поля представляет графику модели, включая блок-схему процесса, анимацию и другие элементы.

Окно свойств модуля служат для настройки параметров модели таких как: время, издержки и другие параметры.

2. Панель основных процессов

2.1. Графические модули

2.1.1. Модуль Create



Рисунок 2 – Модуль Create

Этот модуль является отправной точкой для сущностей в имитационной модели.

Сущности – это индивидуальные элементы, обрабатываемые в системе. Создание сущностей модулем происходит по расписанию, или же основываясь на значении времени между прибытиями сущности в модель. Покидая модуль, сущности начинают обрабатываться в системе. Тип создаваемых сущностей определяется в этом модуле.

Применение:

- прибытие различных документов в сфере бизнеса (например: заказы, чеки, документация);
- прибытие клиентов в сфере обслуживания (например: в ресторан, в магазин);
- начало изготовления продукции на производственной линии.

Таблица 1. Параметры модуля Create

Параметры - Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме

Entity Type - Название типа сущности, который будет создаваться модулем

Type - Способ формирования потока прибытия. Type может иметь значение:

- **Random** (используется экспоненциальное распределение со средним значением, определенным пользователем),
- **Schedule** (определяется модулем Schedule),
- **Constant** (будет использоваться, определенное пользователем, постоянное значение; например, 100) или
- **Expression** (поток прибытия будет формироваться по определенному выражению)

Value - Определяет среднее значение экспоненциального распределения (Random) или постоянное значение времени между прибытиями сущностей (если Type = Constant)

Schedule Name - Имя расписания, которое определяет характер прибытия сущности в систему

Expression - Этот параметр задает тип распределения или выражение, определяющее время между прибытиями сущностей в модель

Units - Единицы измерения времени между прибытиями (день, час, минута, секунда)

Entities per arrival - Количество сущностей входящих в систему за одно прибытие

Max arrivals - Максимальное число сущностей, которое может создать этот модуль

First Creation - Время, через которое придет первая сущность в модель от начала симуляции

2.1.2. Модуль Process

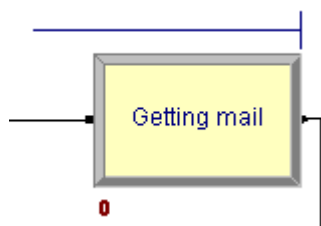


Рисунок 3 – Модуль Process

Этот модуль является основным модулем процесса обработки в имитационной модели. В модуле имеются опции использования ресурсов. Кроме стандартного модуля Process, можно использовать подмодель, придавая ей особую, определенную пользователем, иерархическую

логическую схему. В модуле можно также задавать добавочные стоимостные и временные характеристики процесса обработки сущности.

Наиболее частое применение модуля Process:

- проверка документов;
- выполнение заказов;
- обслуживание клиентов;
- обработка деталей.

Таблица 2. Параметры модуля Process

Параметры - Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме;

Type - Определяет логическую схему модуля.

- **Standard** означает, что логическая схема находится внутри модуля и зависит от параметра Action.
- **Submodel** показывает, что логическая схема будет находиться ниже в иерархической модели. Подмодель может содержать любое количество логических модулей

Action - Тип обработки происходящей внутри модуля.

- **Delay** просто показывает, что процесс занимает какое-то время и не отражает использование ресурсов.
- **Seize Delay** указывает на то, что в этом модуле были размещены ресурсы и будет происходить задержка, ресурсы будут захватываться (то есть будут заняты обработкой сущности), и их освобождение будет происходить позднее.
- **Seize Delay Release** указывает на то, что ресурс(-ы) были захвачены, а затем через время освободились.
- **Delay Release** означает, что ресурсы до этого были захвачены сущностью, а в таком модуле сущность задержится и освободит ресурс.

Все эти параметры доступны только тогда, когда Type = Standard

Priority - Значение приоритета модулей использующих один и тот же ресурс где угодно в модели. Это свойство не доступно, если Action = Delay или Delay Release, или когда Type = Submodel

Resources - Определяет ресурсы или группы ресурсов, которые будут обрабатывать сущности в этом модуле (см. Модуль Process – Ресурсы)

Delay Type - Тип распределения или процедура, определяющая параметры задержки

Units - Единицы измерения времени задержки (день, час, минута, секунда)

Allocation - Определяет стоимостные характеристики обработки.

- **Value Added** - означает учитывать стоимостные характеристики, а
- **Non-Value Added** не учитывать

Minimum - Поле, определяющее минимальное значение для равномерного и треугольного распределения

Maximum - Поле, определяющее максимальное значение для равномерного и треугольного распределения

Value - Поле, определяющее среднее значение для нормального и треугольного распределения или значения для постоянной временной задержки

Std Dev - Параметр, определяющий стандартное отклонение для нормального распределения

Expression - Поле, в котором задается выражение, определяющее значение временной задержки, если Delay Type = Expression

2.1.3. Модуль Decide

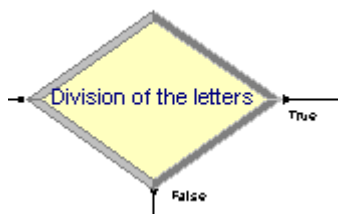


Рисунок 4 – Модуль Decide

Этот модуль позволяет учитывать принятие решений в модели. Он включает опции принятия решений:

- основанных на условии By Condition (например, если тип сущности Car)
- или основанных на вероятности By Chance (например, 75% - true, а 25% - false).

Условия могут быть основаны на значении атрибута Attribute, значении переменной Variable, типе сущности Entity Type или основанные на выражении Expression.

Если поставленное условие не выполняется то, сущности будут покидать модуль через ветку False. Данный модуль позволяет выполнять проверку не только одного условия, но и нескольких. Это достигается с помощью свойства Type; N-way by Chance/by Condition. В зависимости от условия сущность идет по нужной ветке.

Применение:

- разделение дел на срочные дела и несрочные;
- перенаправление недоделанных или сделанных неправильно работ на доработку.

Таблица 3. Параметры модуля Decide

Параметры Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме

Type -Тип принятия решения.

- **By Chance** - выбор направления основывается на вероятности.
- **By Condition** – проверка на выполнение условия
- **Percent True** - Значение, определяющее процент сущностей, который пойдут по направлению True

If - Тип условия, которое будет проверяться на выполнение

Named - Имя переменной, атрибута или типа сущности, который будут проверяться при входе сущности в модуль

Is - Математический знак условия, например больше, меньше, равно и т.д.

Value - Значение, с которым будет сравниваться атрибут или переменная пришедшей сущности. Если тип условия – Expression, то в выражении должен стоять знак условия, например Color<> Red

2.1.4. Модуль Batch

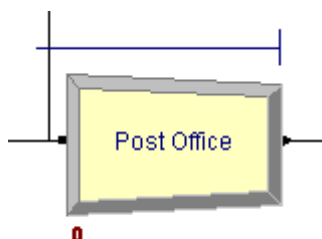


Рисунок 5 – Модуль Batch

Этот модуль отвечает за механизм группировки в имитационной модели.

Группировка может быть постоянной или временной. Временно сгруппированные комплекты позднее могут быть разъединены с помощью модуля Separate.

Комплекты могут состоять из любого числа входящих сущностей, определенного пользователем или же сущности могут объединяться в комплект в зависимости от атрибута сущности. Временные и стоимостные характеристики выходящей сущности, представляющей комплект будут равны сумме характеристик вошедших в группу сущностей.

Сущности прибывают в модуль, становятся в очередь и остаются там до тех пор, пока в модуле не будет набрано заданное количество сущностей. Когда соберется нужное число сущностей создается сущность представляющая комплект.

Применение:

- собрать необходимое количество данных, прежде чем начинать их обработку;
- собрать ранее разделенные копии одной формы;
- соединить пациента и его больничную карту приема к врачу.

Таблица 4. Параметры модуля Batch

Параметры Описание

Name - Уникальное имя модуля, которое будет отражено в блок-схеме

Type - Способ группировки сущностей, может быть

- **Temporary** (временная),
- **Permanent** (постоянная)

Batch Size - Число сущностей, образующих один комплект

Rule - Определяет, по какому признаку будут группироваться.

- Если **Rule = Any Entity**, это значит, что первые 3 (если Batch Size = 3) сущности будут сгруппированы.
- Если **Rule = By Attribute**, то будет объединяться заданное количество сущностей с определенным атрибутом. Например, если Attribute Name = Color, то все сущности, имеющие одинаковое значение атрибута Color, будут сгруппированы

Attribute Name - Имя атрибута, по значению которого будут группироваться сущности

2.1.5. Модуль Separate

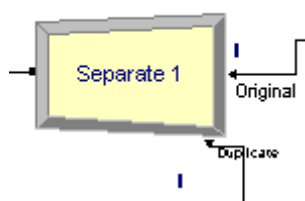


Рисунок 6 – Модуль Separate

Этот модуль может использоваться как для создания копий входящих сущностей, так и для разделения ранее сгруппированных сущностей. Правило для разделения стоимостных и временных характеристик копий сущностей и разделенных сущностей определяется пользователем. Когда временно сгруппированные сущности прибывают в модуль, они раскладываются на составные сущности. Сущности покидают модуль в той же последовательности, в которой они добавлялись в комплект.

Если модуль создает копии сущностей, то пользователь может задать количество дубликатов сущности. У дублированной сущности значения атрибута, а также анимационная картинка такие же, как и оригинала.

Оригинальная сущность также покидает модуль.

Применение:

- разъединение ранее сгруппированных комплектов документов;
- для параллельной обработки счетов и документов по одному заказу.

Таблица 5. Параметры модуля Separate

Параметры - Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме

of Duplic - Количество создаваемых копий входящей сущности

Type - Способ разделение входящей в модуль сущности.

- **Duplicate Original** – просто делает дубликаты входящей сущности.
- **Split Existing Batch** - требует чтобы входящая сущность была предварительно временно сгруппирована
- **Cost to Duplicates** - Разделение стоимостных и временных характеристик входящей сущности между выходящими. Это значение определяется пользователем в процентах, т.е. сколько процентов от стоимостных и временных характеристик входящей сущности уйдет копиям (характеристики между копиями делятся поровну)

Allocation Rule - Метод разделения стоимости и времени, если выбран Type=Split Existing Batch.

- **Retain Original Entity Values** – сохраняет оригинальные значения сущностей.
- **Take All Representative Values** – все сущности принимают одинаковое значение.
- **Take Specific Representative Values** – сущности принимают специфическое значение

2.1.6. Модуль Assign



Рисунок 7 – Модуль Assign

Этот модуль предназначен для задания нового значения переменной, атрибуту сущности типу сущности, анимационной картинке сущности или другой переменной в системе.

В одном модуле можно сделать только одно назначение.

Пример применения модуля Assign:

- установление приоритета для клиентов;
- присвоение номера вышедшему приказу.
-

Таблица 6. Параметры модуля Assign

Параметры - Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме

Type - Тип назначения, которое будет осуществляться. Other может включать в себя встроенные в Арену переменные, такие как вместимость ресурса или конечное время симуляции

Variable Name - Имя переменной, которая будет изменяться в этом модуле

Attribute Name - Имя атрибута, который будет изменяться в этом модуле

Entity Type - Новый тип сущности, присваиваемый сущности в этом модуле

Entity Picture - Новая анимационная картинка для сущности, прошедшей этот модуль

Other - Имя переменной в системе, которая будет меняться

New Value - Присваиваемое новое значение для атрибута, переменной

2.1.7. Модуль Record



Рисунок 8 – Модуль Record

Этот модуль предназначен для сбора статистики в имитационной модели.

Модуль может собирать различные типы статистики, включая время между выходами сущностей из модуля, статистику сущности (время цикла, стоимость), статистику за период времени (период времени от заданной точки до текущего момента). Также доступен количественный тип статистики.

Частое применение модуля:

- подсчитать, какое количество заказов было выполнено с опозданием;
- подсчитать количество работы, совершаемое за один час.

Таблица 7. Параметры Модуль Record

Параметры - Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме

Type - Определяет тип статистики, которая будет собираться. Count – будет увеличивать или уменьшать статистику на заданное значение. Entity Statistics будет собирать общую статистику о сущности, например, время цикла, стоимостные характеристики и т. д. Time Interval будет считать разницу между значением атрибута и текущим временем моделирования. Time Between будет отслеживать время между вхождением сущностей в модуль. Expression будет просто фиксировать значение определяемое выражением

Attribute Name - Имя атрибута, значение которого будет использоваться для интервальной статистики

Value - Значение, которое будет добавляться к статистике, когда в модуль будет прибывать сущность

2.1.8. Модуль Dispose



Рисунок 9 – Модуль Dispose

Этот модуль является выходной точкой из имитационной модели. Статистика о сущности может собираться до того момента пока она не выйдет из системы.

Применение:

- Окончание бизнес процесса;
- Клиенты покидают отдел.

Таблица 8. Параметры модуля Dispose

Параметры Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме

Record Entity Statistics - Определяет, будет ли вестись статистика о выходе сущности из системы

2.2. Модули данных

2.2.1. Модуль *Entity*

Этот модуль определяет тип сущности и ее анимационную картинку в имитационном процессе, также определяет стоимостную информацию.

Для каждого источника должен быть определен тип сущности, который он генерирует.

Применение модуля Entity:

- Документы: факсы, письма, отчеты и т. д.;
- Люди в моделях больницы или магазина.

Таблица 9. Параметры модуля Entity

Параметры - Описание

Entity Type - Название типа сущности

Initial Picture - Графическое представление сущности в начале имитационного процесса. Это значение может быть впоследствии изменено с помощью модуля Assign. Просмотреть анимационные картинки можно так: Edit/ Entity picture

Holding Cost/Hour - Почасовая стоимость обработки сущности в системе. Эта стоимость учитывается, когда сущность находится в системе, либо в очереди, либо в стадии обработки

Initial VA Cost - Значение, присваиваемое атрибуту сущности «добавочная стоимость». Значение атрибута будет увеличиваться каждый раз, как только сущность будет обрабатываться процессом с добавочной стоимостью

Initial NVA Cost - Значение, присваиваемое атрибуту сущности «не добавочная стоимость». Значение атрибута будет увеличиваться каждый раз, как только сущность будет обрабатываться процессом с не добавочной стоимостью

2.2.2. Модуль *Queue*

Этот модуль данных предназначен для изменения правила расстановки сущностей в очереди. По умолчанию тип очереди First in First out.

Применение:

- Стопка документов, ожидающих освобождения ресурса;
- Место для собирания частей ожидающих упаковки (группировки).

Таблица 10 – Параметры модуля Queue

Параметры - Описание

Name - Уникальное имя модуля, которое будет отражено в блок схеме Attribute Name Имя атрибута, значение которого будет учитываться, если тип = Lowest Attribute Value или Highest Attribute Value

Type - Правило расстановки сущностей в очереди. First in First out – первый вошел, первый вышел. Last in first out – последний пришел, первый вышел. Lowest Attribute Value – первый выйдет из очереди тот, значение атрибута у которого ниже. Highest Attribute Value – первый выйдет из очереди тот, значение атрибута у которого наивысшее

2.2.3. Модуль *Resource*

Этот модуль предназначен для определения ресурсов и их свойств в имитационном процессе, кроме того, модуль включает в себя стоимостную информации о ресурсах и вместимость ресурсов. Ресурсы могут иметь фиксированную вместимость или же основанную на расписании. У ресурсов с фиксированной вместимостью в течение имитационного процесса вместимость изменяться не может.

Применение:

- Люди (клерки, продавцы, бухгалтеры, рабочие и т. д.);
- Оборудование (телефонная линия, станок, компьютер).

Таблица 11 – Параметры модуля Resource

Параметры - Описание

Name - Имя ресурса

Type - Метод, определяющий вместимость ресурса. Fixed Capacity – фиксированная вместимость ресурса. Based on Schedule – вместимость ресурса определяется модулем Schedule

Capacity - Число ресурсов, находящихся в системе

Schedule Name - Имя Schedule модуля, который определяет вместимость ресурса, если Type = Based on Schedule

Busy / Hour - Почасовая стоимость обработки сущности ресурсом. Время учитывается только тогда, когда ресурс занят обработкой и прекращает учитываться, когда ресурс освобождается

Idle / Hour - Стоимость ресурса, когда он не занят

Per Use - Стоимость обработки ресурсом одной сущности (не зависит от времени)

2.2.4. Модуль Schedule

Этот модуль может использоваться вместе с модулем Resource для определения вместимости ресурса. Также модуль используется вместе с модулем Create для задания расписания прибытия сущностей.

Применение:

- Расписание работы персонала с перерывами на обед;
- Значение покупателей прибывающих в супермаркет.

Таблица 12 – Параметры модуля Schedule

Параметры - Описание

Name - Название расписания

Type - Тип расписания, который может быть Capacity (расписание для ресурсов), Arrival (для модуля Create) или Other (разнообразные временные задержки или факторы)

Time Units - Масштаб оси времени в графике расписания

2.2.5. Модуль Set

Этот модуль данных, который описывает группу ресурсов, использующихся в модуле Process. В группе могут находиться несколько ресурсов. Модуль set автоматически создает ресурсы, вместимость которых по умолчанию равна 1 и без всякой стоимостной информации. Следовательно, если для ресурсов входящих в группу не нужно стоимостной информации и вместимость более 1, то можно обойтись созданием только модуля Set.

Возможно применение модуля для организации работы группы работников, например, по очереди.

Таблица 13 – Параметры модуля Set

Параметры - Описание

Name - Название группы

Members - Перечисляет ресурсы, входящие в группу. Порядок перечисления ресурсов важен, когда в модуле Process используется правило выбора Cyclical или Preferred Order

Resource Name - Названия ресурсов входящих в группу

2.2.6. Модуль Variable

Этот модуль данных определяет значение переменной. Переменные, относящиеся к модулю Decide или Assign, могут использоваться в выражениях. Если переменная не описана в этом модуле, то ее значение равно 0.

Применение:

- Число документов обрабатываемых в час;
- Присвоение серийного номера для идентификации продукции.

Таблица 14 – Параметры модуля Variable

Параметры - Описание

Name - Имя переменной

Initial Value - Первоначальное значение переменной. Это значение в последствии может меняться модулем Assign

Rows - Число строк в размерной переменной

Columns - Число столбцов в размерной переменной

Clear Option - Определяет время, когда значение переменной сбрасывается в начальное значение.

Statistics – сбрасывает переменную в начальное значение в любой момент, когда статистика была расчищена.

System – сбрасывает переменную в начальное значение в любой момент, когда система была расчищена.

None – никогда не сбрасывает переменную в начальное значение, исключая предшествующую первой репликации

Statistics - Определяет, будет ли вестись статистика по этой переменной

3. Панель отчетов

С помощью панели отчетов можно просмотреть результаты проигрывания имитационной модели. На панели отчетов представлены несколько видов отчетов. Отчет «Краткий обзор категорий» и отчеты по четырем категориям, такие как Сущности, Процессы, Очереди и Ресурсы.

Отчет Category Overview категорий (Краткий обзор категорий)

Отчет Category Overview отражает итоговую информацию о сущностях, процессах, очередях и ресурсах. Также показывает информацию о заданных пользователем переменных и информацию, собранную модулем Record.

Отчет о сущностях

Отчет о сущностях разделен на несколько частей.

Cycle Time: в этой части отчета показано среднее, максимальное и минимальное время существования сущности. Время существования сущности считается с момента её прибытия в систему и до того момента, когда сущность попадает в модуль Dispose. Ниже представляется гистограмма среднего времени цикла для каждого типа сущности.

NVA Cost: в этой части показано среднее, максимальное и минимальное значение не добавочной стоимости сущностей по каждому типу сущностей. Не добавочная стоимость рассчитывается на основании значения NVA Time. **Total Cost:** в этой части показано среднее, максимальное и минимальное значение общей стоимости сущностей по каждому типу сущностей. Общая стоимость вычисляется путем сложения стоимости ожидания, добавочной стоимости и не добавочной стоимости для каждой сущности. Ниже представляется сравнительные гистограммы средних значений общей стоимости сущности для каждого типа сущности.

VA Cost: в этой части показано среднее, максимальное и минимальное значение добавочной стоимости сущностей по каждому типу сущностей. Добавочная стоимость рассчитывается на основании VA Time.

Wait Cost: в этой части показано среднее, максимальное и минимальное значение стоимости ожидания сущностей по каждому типу сущностей. Стоимость ожидания подсчитывается исходя из времени ожидания, стоимости ресурса и стоимостью нахождения сущности в системе.

Wait Time: в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей. Время ожидания – это период времени с момента поступления сущности в очередь (либо в модуле Process ожидает ресурс, либо в модуле Batch ожидает группировки) и до момента выхода из нее (начнет обрабатываться, либо будет сгруппирована).

WIP (Work In Process): в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей.

Отчет о процессах

Отчет о процессах разделен на несколько частей.

Cycle Time: в этой части отчета показано среднее, максимальное и минимальное время цикла процесса. Время цикла процесса считается с момента прибытия в модуль Process и до того момента, когда сущность покидает модуль. В разделе представляется гистограмма среднего времени цикла для каждого процесса.

NVA Cost: в этой части показано среднее, максимальное и минимальное значение недобавочной стоимости сущностей по процессу. Не добавочная стоимость рассчитывается на основании NVA Time. **Total Cost:** в этой части показано среднее, максимальное и минимальное значение общей стоимости сущностей по процессу. Общая стоимость вычисляется путем сложения стоимости ожидания, добавочной стоимости и не добавочной стоимости для каждой сущности. В разделе представляются сравнительные гистограммы средних значений общей стоимости сущности для каждого типа сущности.

VA Cost: в этой части показано среднее, максимальное и минимальное значение добавочной стоимости сущностей по каждому типу сущностей. Добавочная стоимость рассчитывается на основании VA Time.

Wait Cost: в этой части показано среднее, максимальное и минимальное значение стоимости ожидания сущностей по каждому типу сущностей. Стоимость ожидания подсчитывается исходя из времени ожидания, стоимости ресурса и стоимостью нахождения сущности в системе.

Wait Time: в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей. Время ожидания – это период времени с момента поступления сущности в очередь (либо в модуле Process ожидает ресурс, либо в модуле Batch ожидает группировки) и до момента выхода из нее (начнет обрабатываться, либо будет сгруппирована).

WIP: в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей.

Раздел «Ресурсы» содержит информацию о ресурсах.

Раздел «Очереди» содержит информацию о очередях.

4. Панель навигации

С помощью панели навигации можно быстро передвигаться по различным уровням модели, быстро менять виды. Можно задать быстрые клавиши для изменения вида. Виды подмоделей создаются автоматически, но также возможно добавить новые виды с помощью команды Add View. Можно передвигаться не только по различным уровням модели, но также быстро получать нужный масштаб какой – либо части модели.

5. Построитель выражений

ПП Arena позволяет строить сложные выражения. Это достигается с помощью Expression Builder. Построитель выражений имеет внешний вид, показанный на рисунке 10.

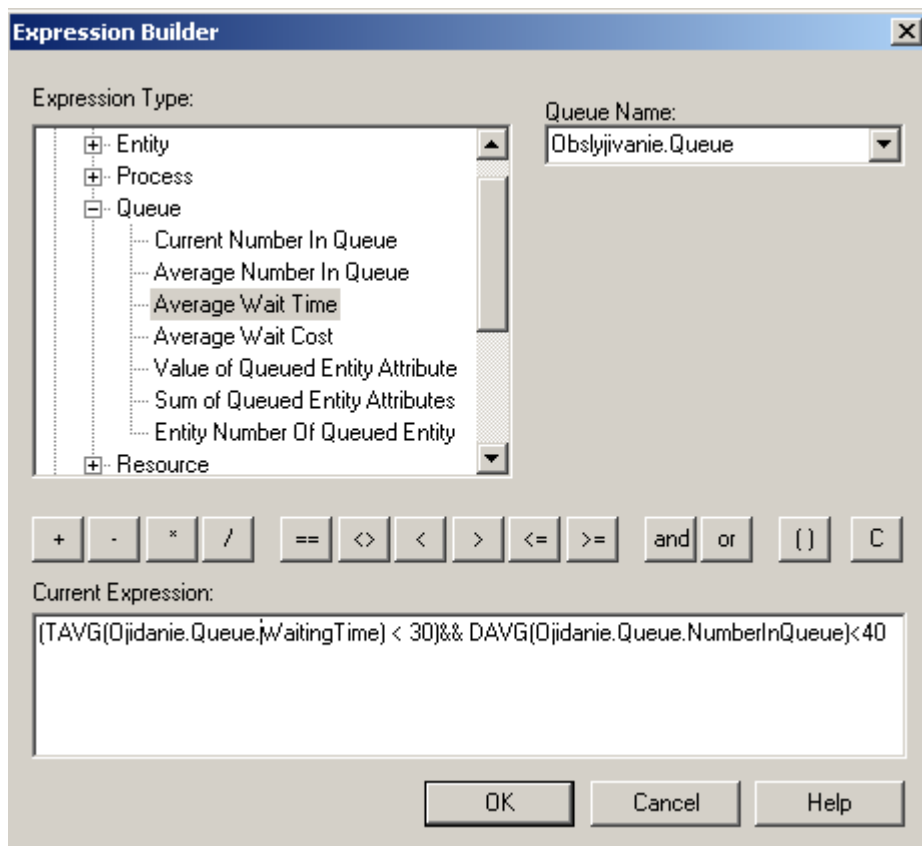


Рисунок 10 – Внешний вид построителя выражений

Построитель выражений имеет 3 секции:

- 1 – окно возможных выражений элементов, организованных по типам;
- 2 – окно аргументов для выделенных выражений;
- 3 – окно выражений.

Также имеются кнопки операторов (сложение, вычитание, и т.д.; элементы сравнения, логические операторы и т.д.)

Приложение 2. Пример исследования имитационной модели управления объектом средствами системы Arena.

