

Доступ к панели инструментов диаграммы

1. Вкладка Design -> поле Diagram -> Кнопка Toolbox
2. Ctrl+shift+3

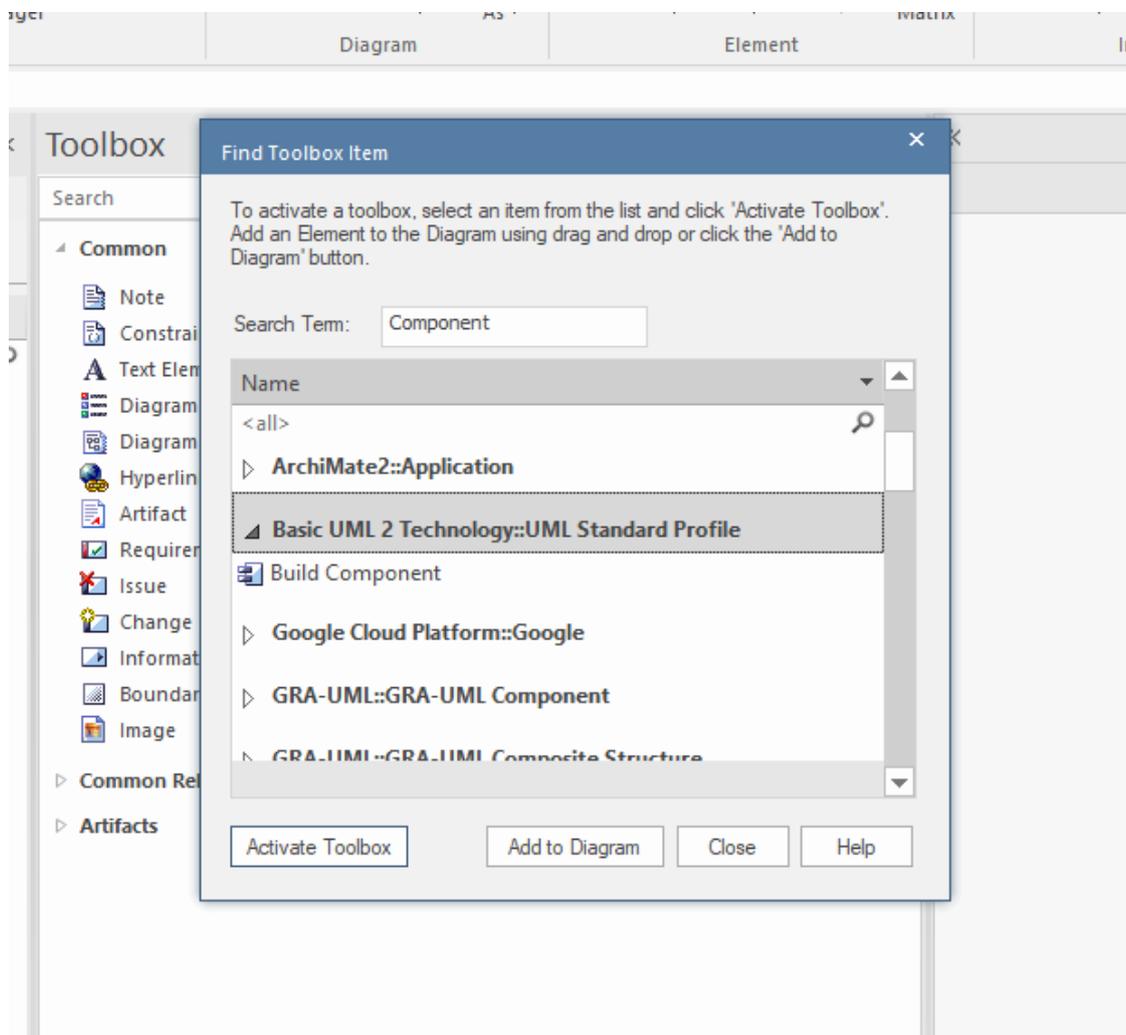
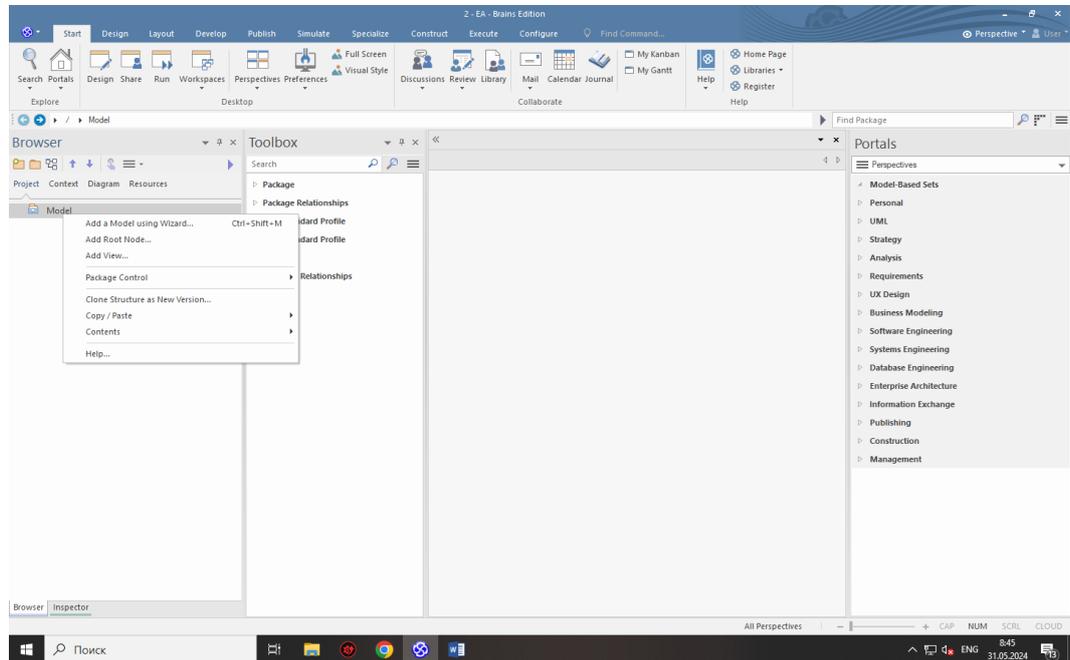
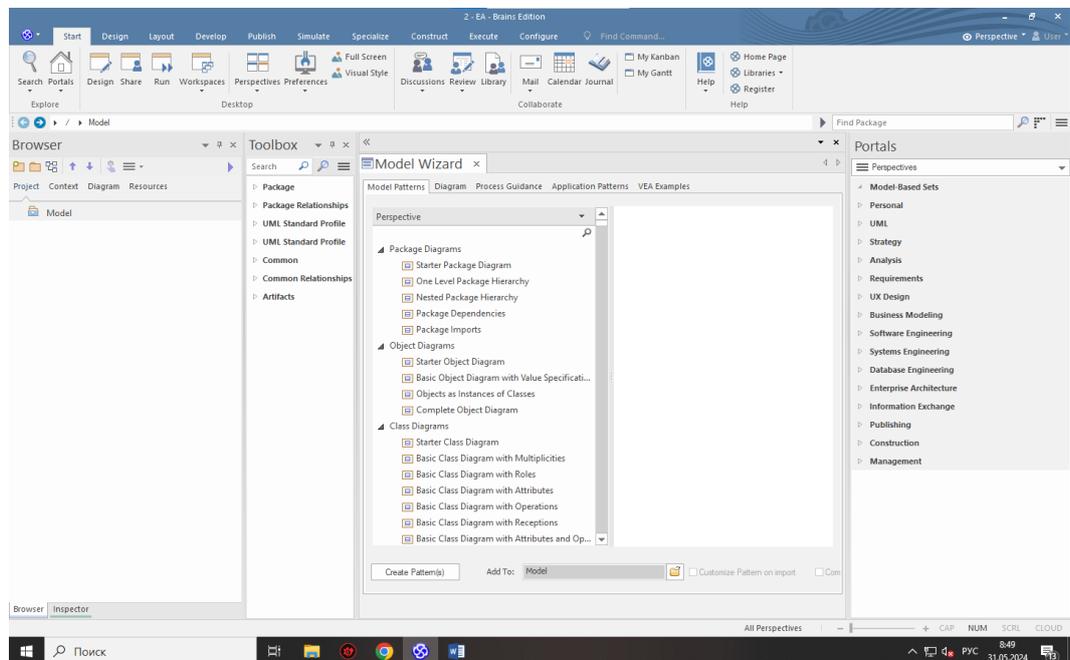


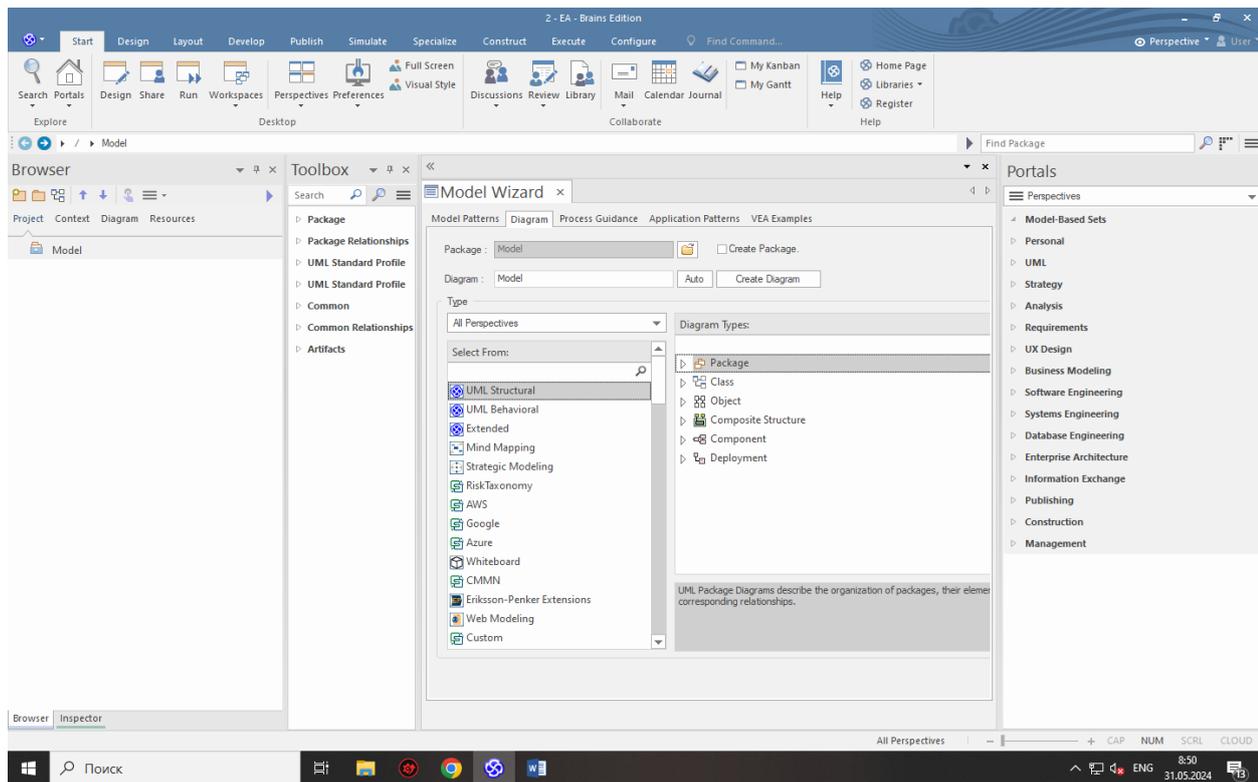
Диаграмма пакетов (Diagram package)

В окне под названием Browser, находящейся в левой части программы, по созданной пользователем модели нажимаем ПКМ. Вылезает список, где необходимо выбрать «Add a Model using Wizard» (Добавить модель с помощью мастера).



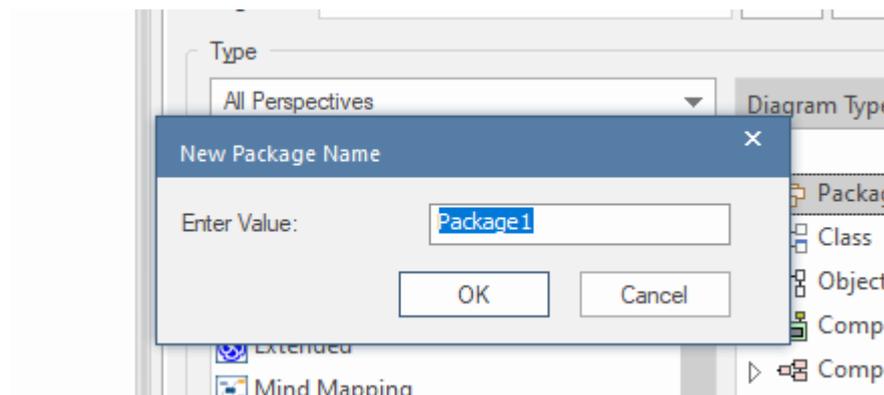
Далее откроется окно «Model Wizard» (Мастер моделей), где необходимо перейти во вкладку Diagram.



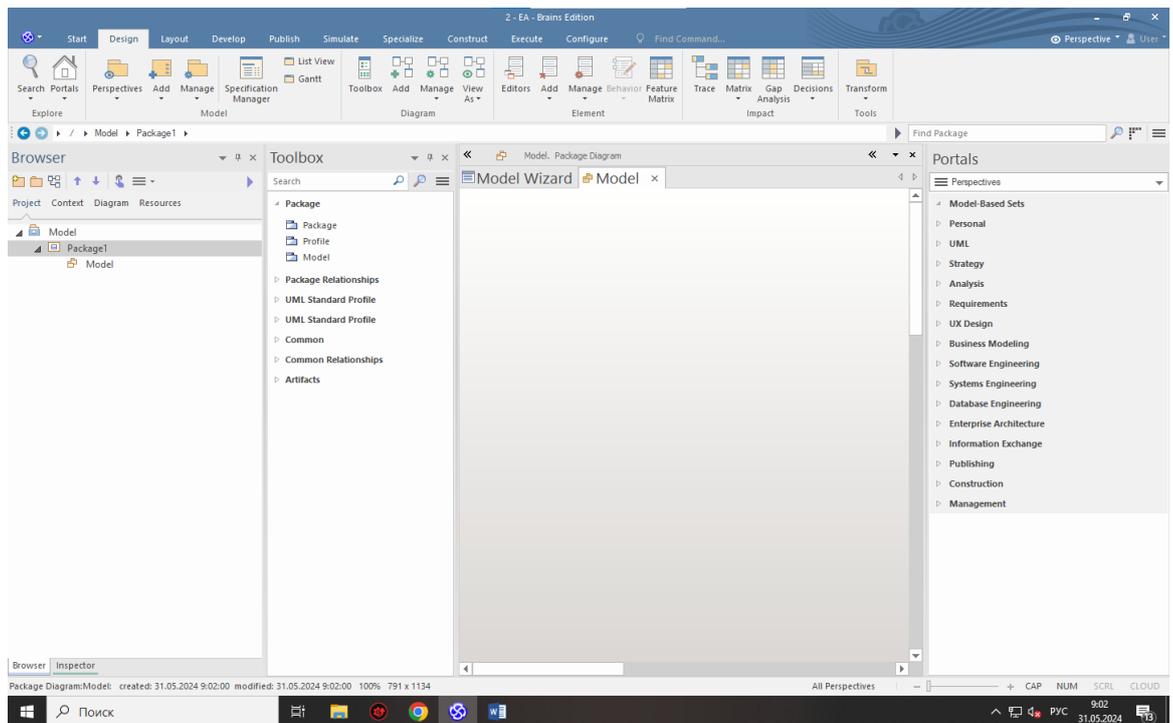


В левом окне списков стандартов необходимо выбрать «UML Structural», после чего, указать, какой тип диаграммы нужно создать (в данном случае, пакетов). Также для создания пакета необходимо нажать на поле для галочки «Create Package». После этого нажимаем кнопку «Create Diagram».

После появления окна «New Package Name», в поле ввода необходимо ввести название среды для работы с пакетами. После этого нажимаем кнопку «OK».



В левом окне «Browser» необходимо выбрать созданный пакет, после чего в среднем окне появится вкладка с названием созданной пользователем модели. В панели с названием «Toolbox» необходимо выбрать вариант пакета, который необходимо спроектировать.



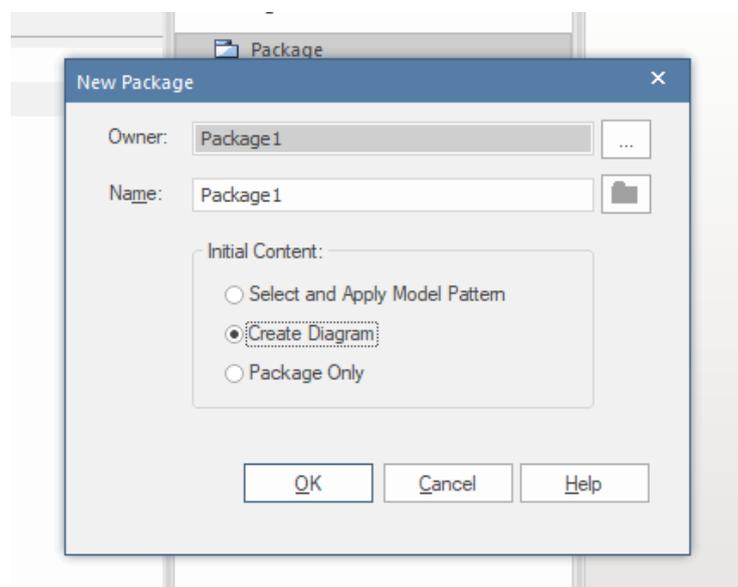
ЛКМ зажимаем на выбранный пакет, и перетаскиваем в основное поле работы с диаграммами. Появится окно «New Package», где необходимо выбрать имя пакета и исходное содержание (Initial Content).

Из предложенного:

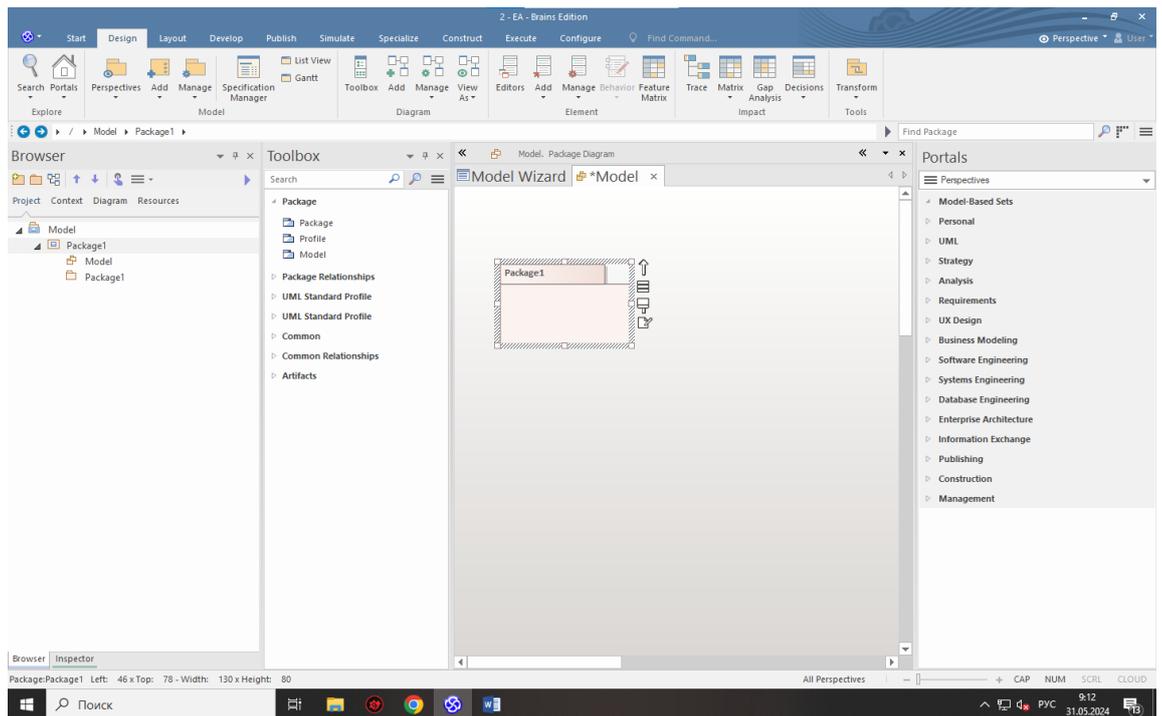
- выберите и примените шаблон модели (Select and Apply Model Pattern);
- создать диаграмму (Crate Diagram);
- только пакет (Package only).

(В данной инструкции выбрано только пакет).

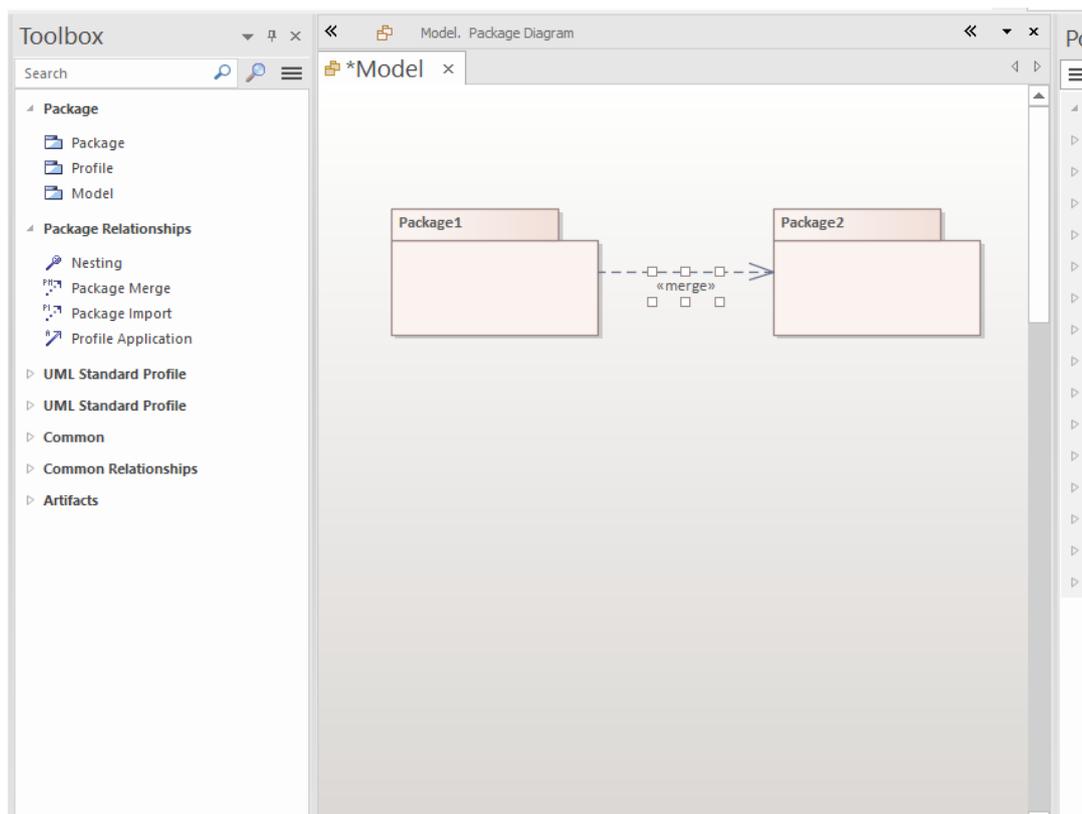
После этого нажимаем кнопку «ОК».



Как показано на картинке ниже, появляется пакет с названием, заданным пользователем.



Для того чтобы выбрать связь между пакетами необходимо развернуть в окне «Toolbox» вкладку Package relationships. Выбираем нужную связь и перетаскиваем и ждем ЛКМ на пакет, после чего нужно отредактировать её по усмотрению пользователя.



Значки панели инструментов элемента диаграммы пакета

Иконка	Описание
 Package	Пакеты используются для организации содержимого вашего проекта, но при добавлении на диаграмму они могут быть использованы для отображения структуры и взаимосвязей вашей модели.
 Profile	Создает пакет профилей, который имеет стереотип «profile» на диаграмме пакетов в вашей модели технической разработки. Пакет профилей используется для определения новых типов структур в модели.
 Model	Генерирует пакет модели со стереотипом «model» для представления родительского узла в структуре модели.

Значки панели инструментов взаимосвязей диаграммы пакетов

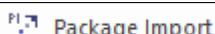
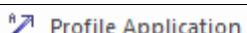
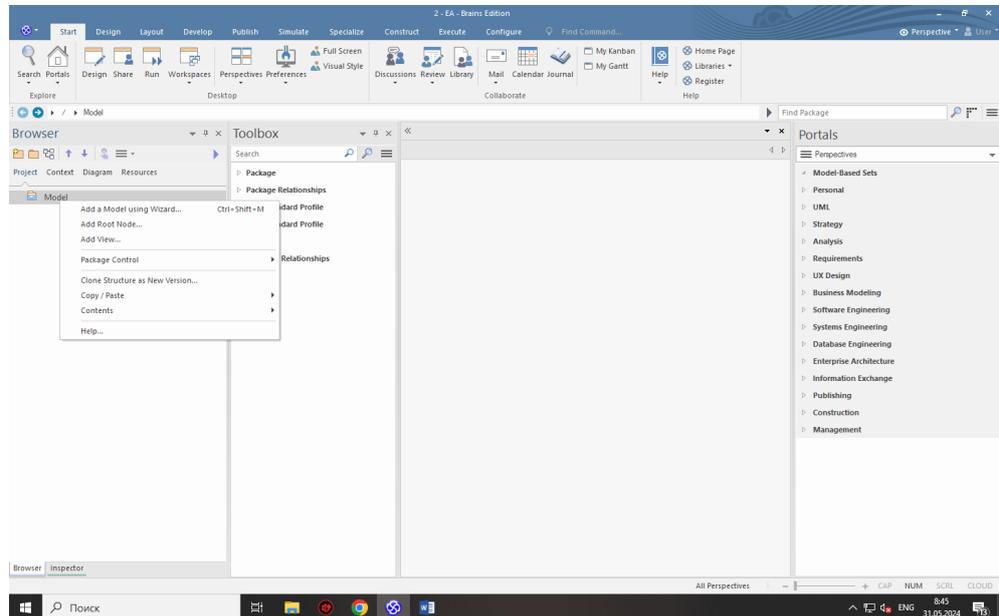
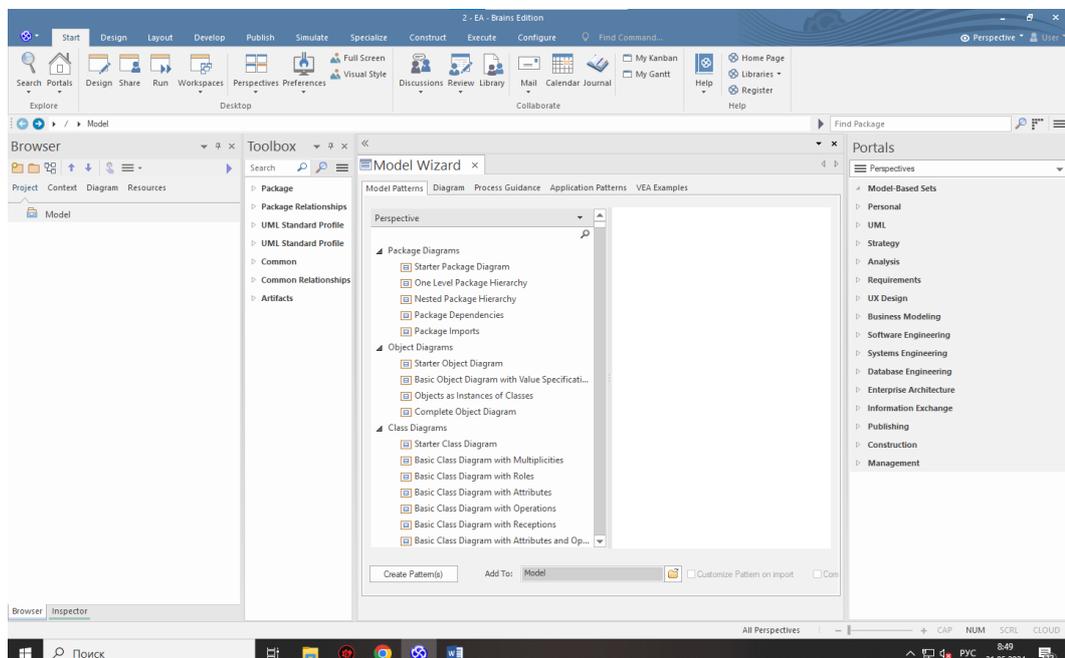
Иконка	Описание
 Nesting	<p>Соединитель вложенности — это альтернативное графическое обозначение для выражения включения или вложенности элементов в другие элементы.</p> <p>Содержимое пакета можно просмотреть, щелкнув фон диаграммы, чтобы отобразить диалоговое окно «Свойства» диаграммы, выбрав вкладку «Элементы» и установив флажок «Содержимое пакета» на панели «Показать отсеки».</p>
 Package Merge	На диаграмме пакета объединение пакетов указывает на связь между двумя пакетами, при которой содержимое целевого пакета было объединено с содержимым исходного пакета.
 Package Import	<p>Отношения «Импорт пакета» создаются из исходного пакета в пакет, содержимое которого было импортировано.</p> <p>Соединитель «импорт» указывает, что элементы целевого целочисленного пакета, который в этом примере представляет собой один целый класс, были импортированы в контроллер пакета.</p> <p>Пространство имен контроллера получает доступ к классу Integer; пространство имен Integer не затрагивается.</p>
 Profile Application	Связь «Приложение профиля» указывает, что исходный профиль был применен к целевому пакету.

Диаграмма развертывания (Deployment Diagram)

В окне под названием Browser, находящейся в левой части программы, по созданной пользователем модели нажимаем ПКМ. Вылезает список, где необходимо выбрать «Add a Model using Wizard» (Добавить модель с помощью мастера).

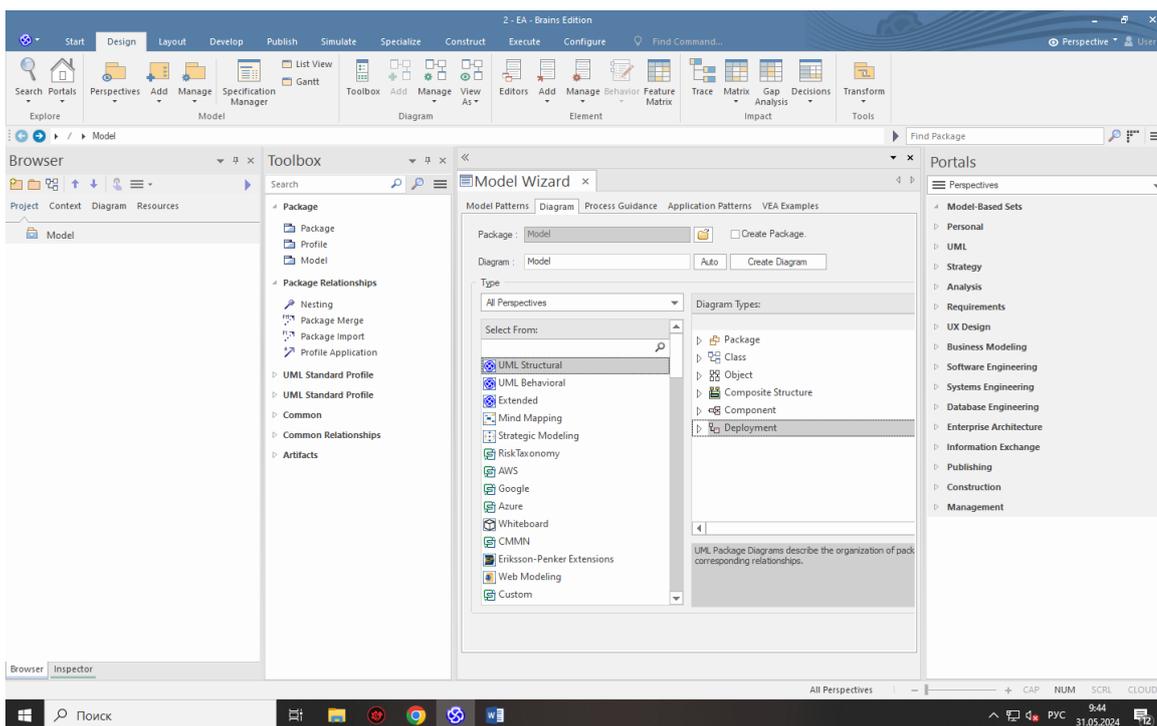


Далее откроется окно «Model Wizard» (Мастер моделей), где необходимо перейти во вкладку Diagram.

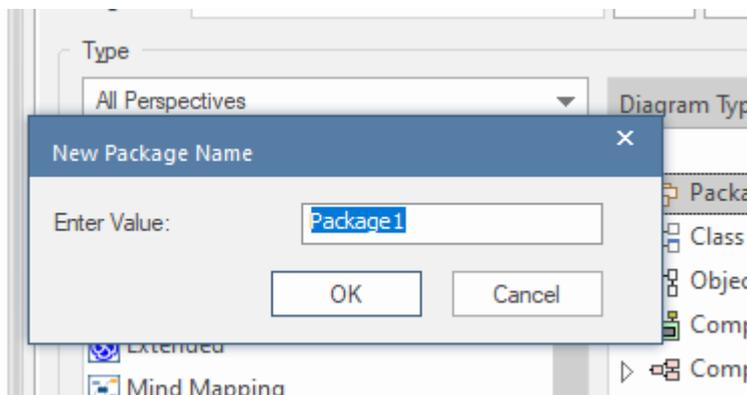


В левом окне списков стандартов необходимо выбрать «UML Structural», после чего, указать, какой тип диаграммы нужно создать (в данном случае, размещения).

Также для создания среды диаграммы необходимо нажать на поле для галочки «Create Package». После этого нажимаем кнопку «Create Diagram».



После появления окна «New Package Name», в поле ввода необходимо ввести название среды для работы с диаграммой. После этого нажимаем кнопку «ОК».



В левом окне «Browser» необходимо выбрать созданный пакет, после чего в среднем окне появится вкладка с названием созданной пользователем модели. В панели с названием «Toolbox» необходимо выбрать варианты, которые необходимо спроектировать.

2 - EA - Brains Edition

Start Design Layout Develop Publish Simulate Specialize Construct Execute Configure Find Command...

Search Portals Perspectives Add Manage Specification Manager List View Gantt Toolbox Add Manage View As Editors Add Manage Behavior Feature Matrix Trace Matrix Gap Analysis Decisions Transform

Explore Model Package1

Browser

- Model
- Package1
 - Model
 - Class1
 - Interface1
 - Node1
 - Component1
 - «File» Artifact1
 - DeploymentSpecification1
 - Device1
 - ExecutionEnvironment1

Toolbox

- Node
- Device
- Execution Environment
- Component
- Interface
- Artifact
- Deployment Specification
- Deployment Relationships
 - Associate
 - Communication Path
 - Association Class
 - Generalize
 - Realize
 - Deployment
 - Manifest
- UML Standard Profile
 - Executable
 - File
 - Library
 - Script
 - Source
- UML Standard Profile
- Common
- Common Relationships
- Artifacts

Model Wizard *Model x

Portals

- Model-Based Sets
- Personal
- UML
- Strategy
- Analysis
- Requirements
- UX Design
- Business Modeling
- Software Engineering
- Systems Engineering
- Database Engineering
- Enterprise Architecture
- Information Exchange
- Publishing
- Construction
- Management

Deployment Diagram: Model created: 31.05.2024 9:45:42 modified: 31.05.2024 9:47:02 100% 791 x 1134

All Perspectives

9:47 31.05.2024

Значки панели инструментов элемента диаграммы развертывания

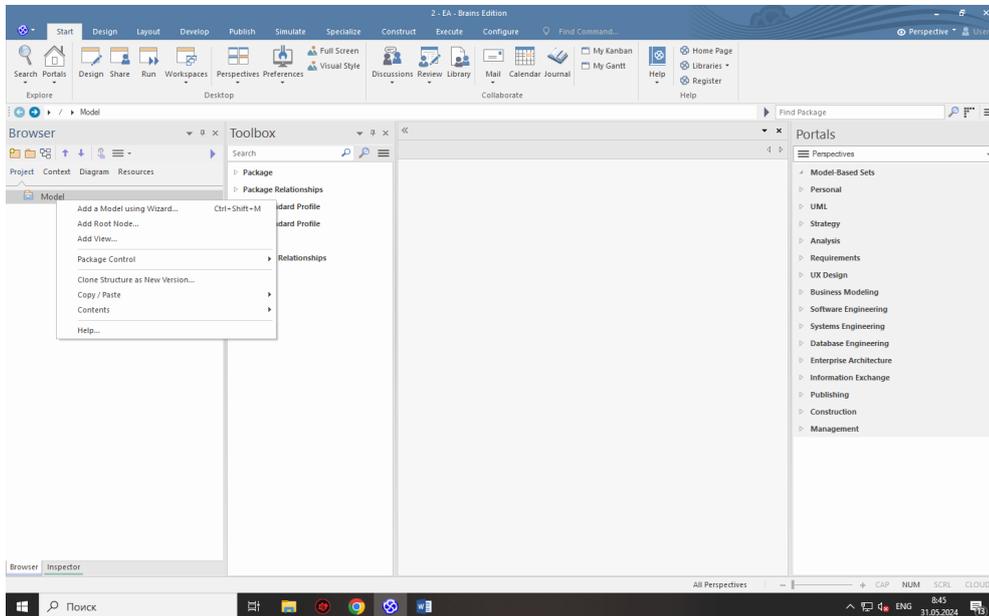
Иконка	Описание
 Node	Узел — это физическое оборудование, на котором развернута система, например сервер рабочей группы или рабочая станция.
 Device	Устройство — это физический электронный ресурс с возможностью обработки, на котором Артефакты могут быть развернуты для выполнения, как показано на диаграмме развертывания.
 Execution Environment	Среда выполнения — это узел, который предлагает среду выполнения для определенных типов компонентов, которые развернуты на нем в форме исполняемых артефактов.
 Component	Компонент — это модульная часть системы, поведение которой определяется предоставляемыми и необходимыми интерфейсами.
 Interface	Интерфейс — это спецификация поведения (или контракта), которую разработчики соглашаются соблюдать.
 Artifact	Артефакт — это любая физическая часть информации, используемая или создаваемая системой.
 Deployment Specification	Спецификация развертывания (спецификация) определяет параметры, определяющие развертывание артефакта, что необходимо для большинства аппаратных и программных технологий.

Значки панели инструментов соединителя схемы развертывания

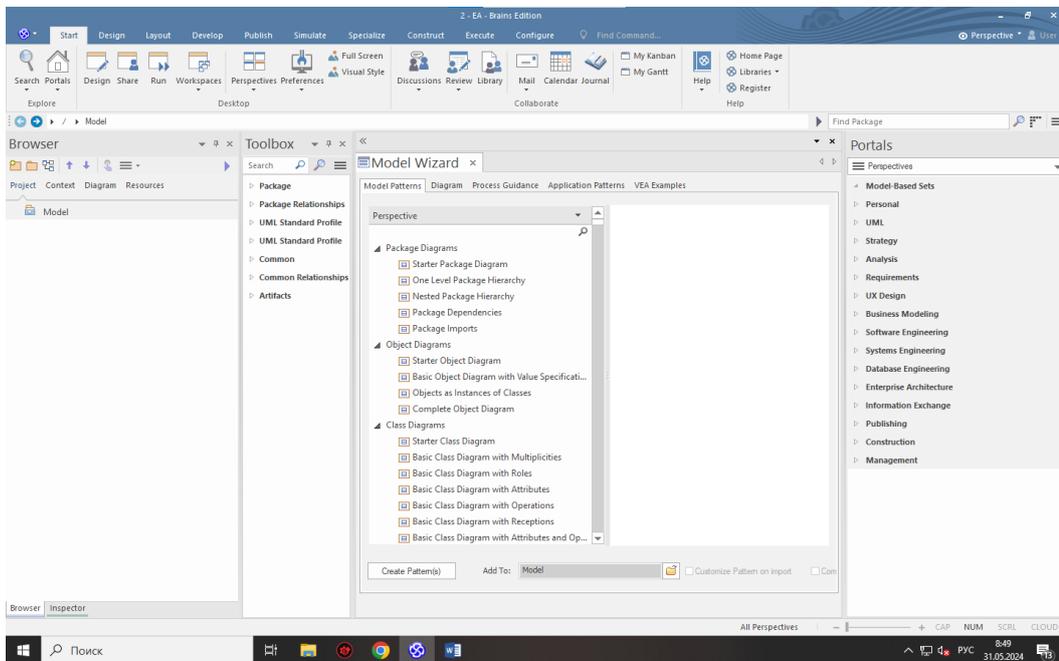
Иконка	Описание
 Associate	Ассоциация подразумевает, что два элемента модели имеют связь, обычно реализуемую как переменную экземпляра в одном или обоих классах.
 Communication Path	Путь связи определяет путь, по которому два DeploymentTargets могут обмениваться сигналами и сообщениями.
 Association Class	Класс ассоциации — это конструкция UML, которая позволяет ассоциации иметь атрибуты и операции (функции).
 Generalize	Обобщение используется для обозначения наследования.
 Realize	Исходный объект реализует или реализует свой целевой объект.
 Deployment	Развертывание — это тип отношения зависимости, который указывает на развертывание артефакта на узле или исполняемом целевом объекте, обычно на диаграмме развертывания.
 Manifest	Отношение Манифест указывает, что источник Артефакта воплощает элемент целевой модели, обычно на диаграммах компонентов и развертывания.

Диаграмма объектов (Diagram object)

В окне под названием Browser, находящейся в левой части программы, по созданной пользователем модели нажимаем ПКМ. Вылезает список, где необходимо выбрать «Add a Model using Wizard» (Добавить модель с помощью мастера).

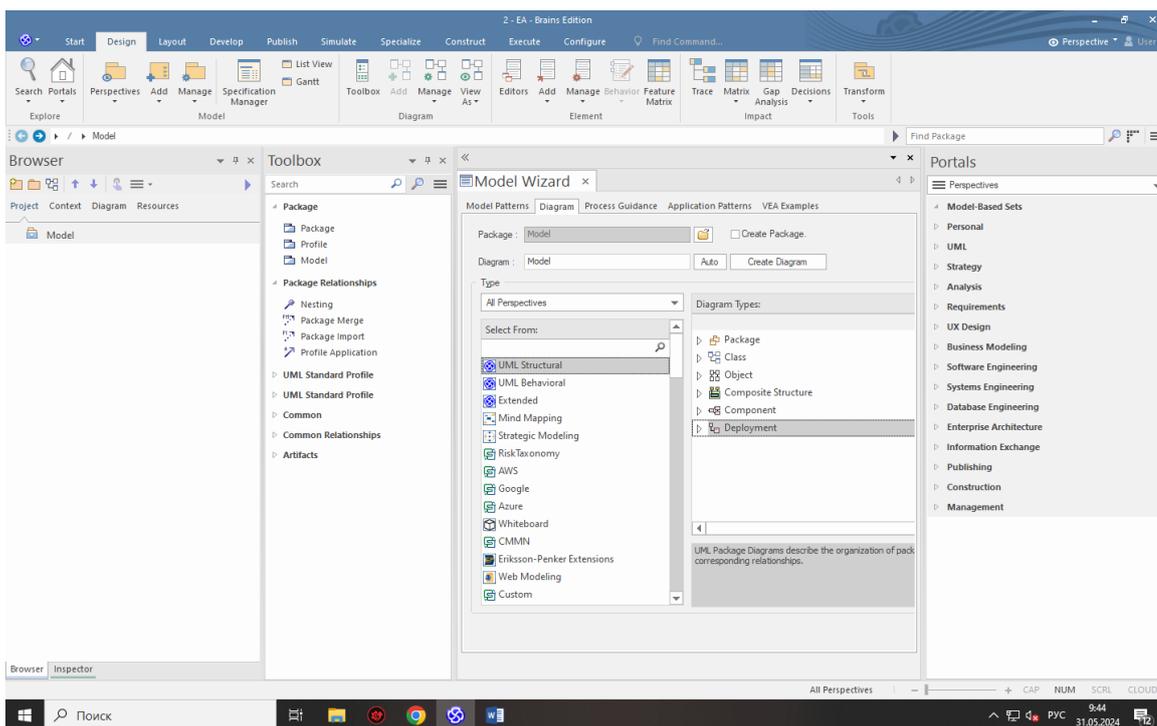


Далее откроется окно «Model Wizard» (Мастер моделей), где необходимо перейти во вкладку Diagram.

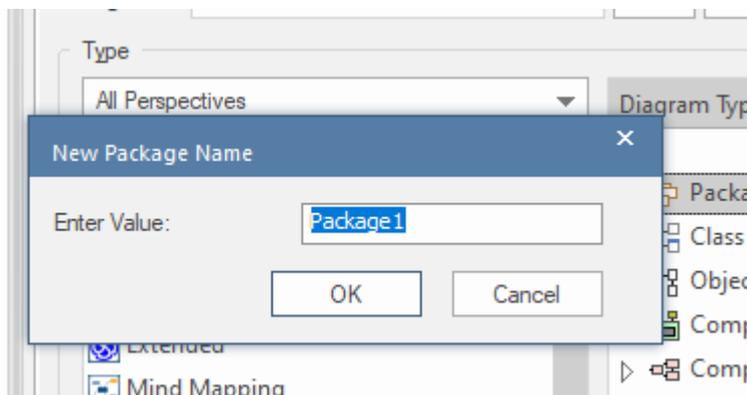


В левом окне списков стандартов необходимо выбрать «UML Structural», после чего, указать, какой тип диаграммы нужно создать (в данном случае, объектов). Также

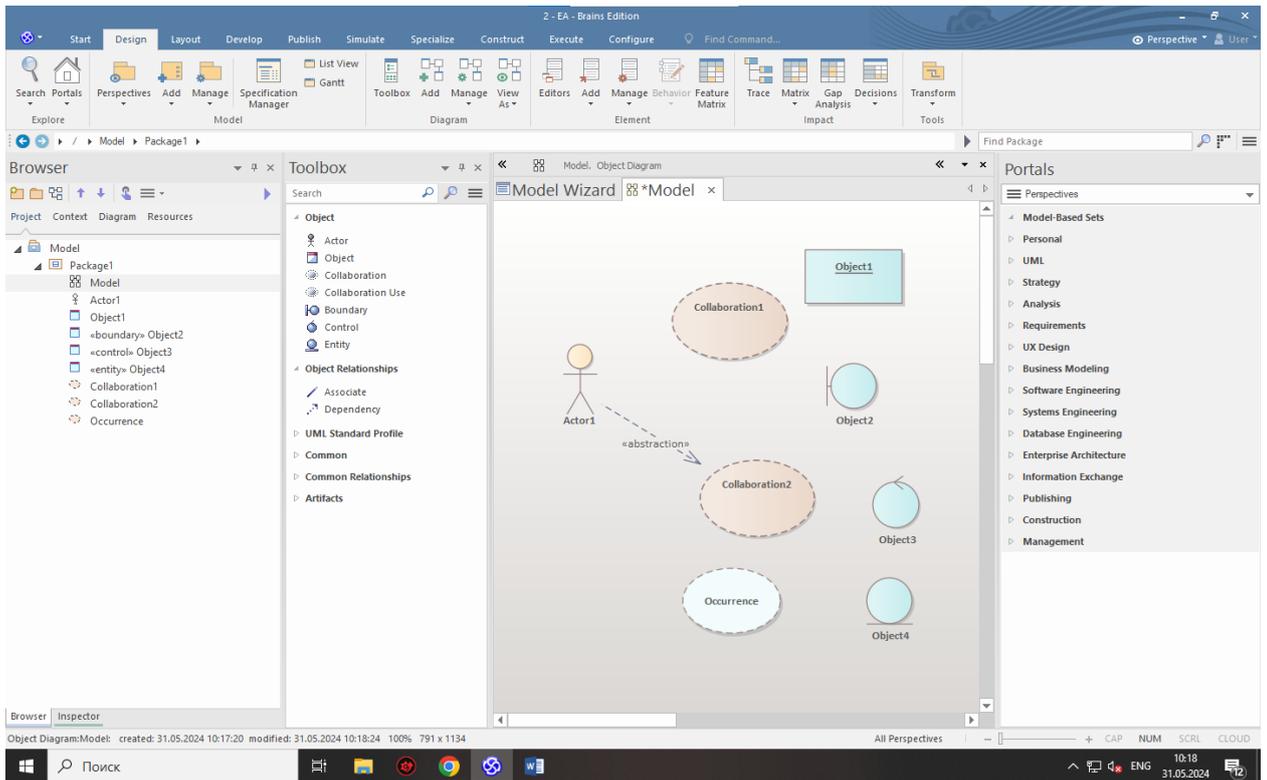
для создания среды диаграммы необходимо нажать на поле для галочки «Create Package». После этого нажимаем кнопку «Create Diagram».



После появления окна «New Package Name», в поле ввода необходимо ввести название среды для работы с диаграммой. После этого нажимаем кнопку «ОК».



В левом окне «Browser» необходимо выбрать созданный пакет, после чего в среднем окне появится вкладка с названием созданной пользователем модели. В панели с названием «Toolbox» необходимо выбрать варианты объектов, которые необходимо спроектировать.



Значки панели инструментов элемента диаграммы объекта

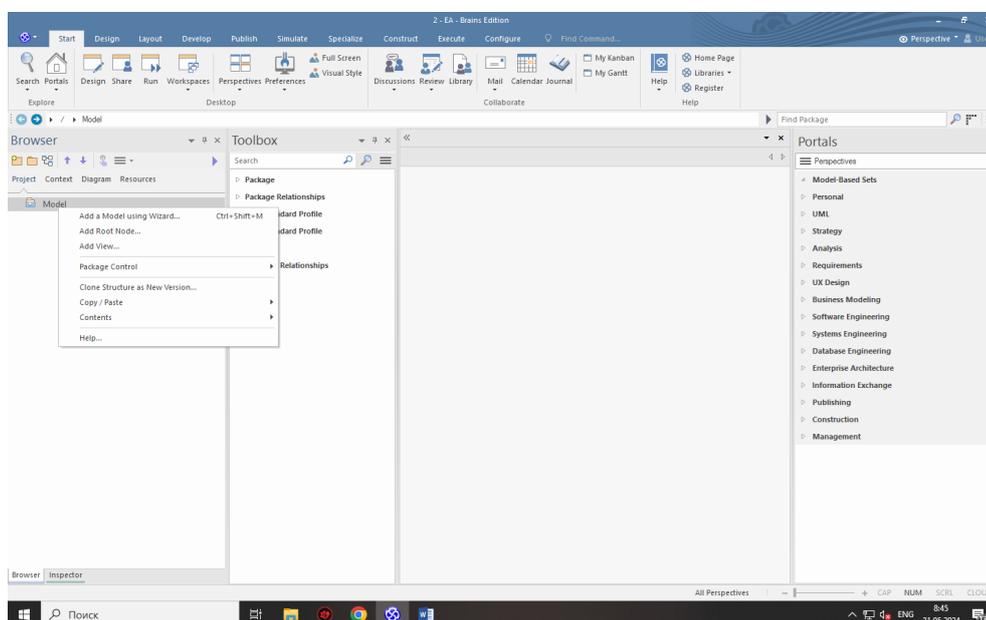
Иконка	Описание
 Actor	Актер — пользователь системы; Пользователь может означать пользователя-человека, машину или даже другую систему или подсистему в модели.
 Object	Объект — это конкретный экземпляр класса во время выполнения.
 Collaboration	Сотрудничество определяет набор взаимодействующих ролей и их соединителей.
 Collaboration Use	Использование сотрудничества. Используйте для применения шаблона, определенного сотрудничеством, к конкретной ситуации на диаграмме составной структуры .
 Boundary	Граница — это стереотипный объект, который моделирует некоторую границу системы, обычно экран пользовательского интерфейса.
 Control	Элемент управления — это стереотипный Объект, который моделирует контролируемую сущность или менеджера.
 Entity	Сущность — это стереотипный объект, который моделирует механизм хранения или сохранения, который фиксирует информацию или знания в системе.

Значки панели инструментов соединителя диаграмм объектов

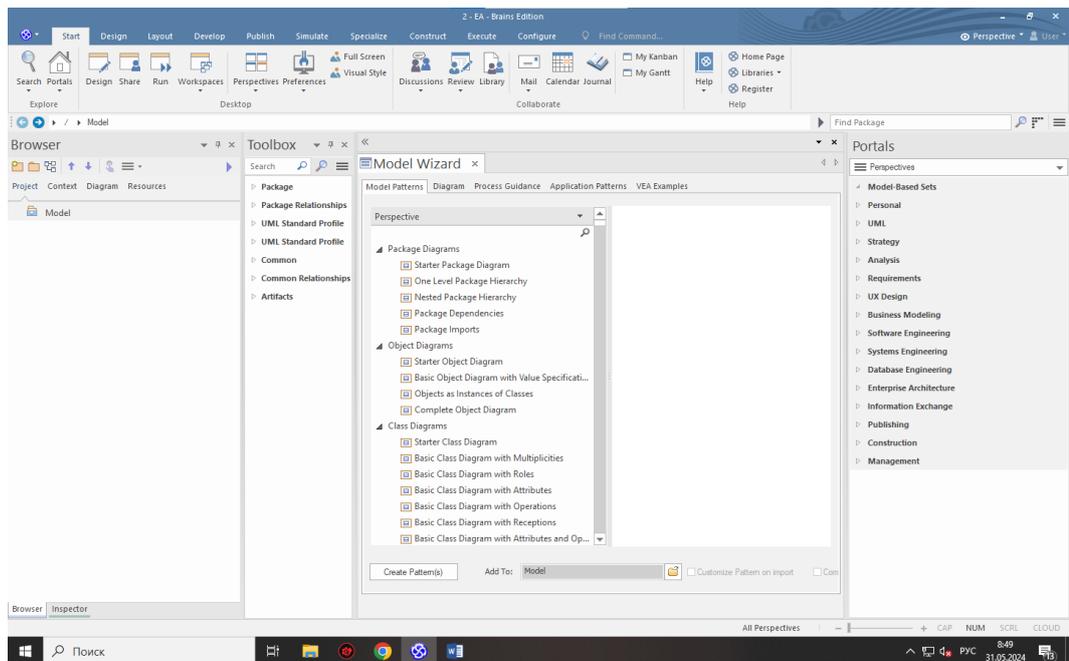
Иконка	Описание
 Associate	Ассоциация подразумевает, что два элемента модели имеют связь, обычно реализуемую как переменную экземпляра в одном или обоих классах.
 Dependency	Отношения зависимости используются для моделирования широкого спектра зависимых отношений между элементами модели на диаграммах вариантов использования, действий и структурных диаграммах и даже между самими моделями.

Диаграмма компонентов (Component Diagram)

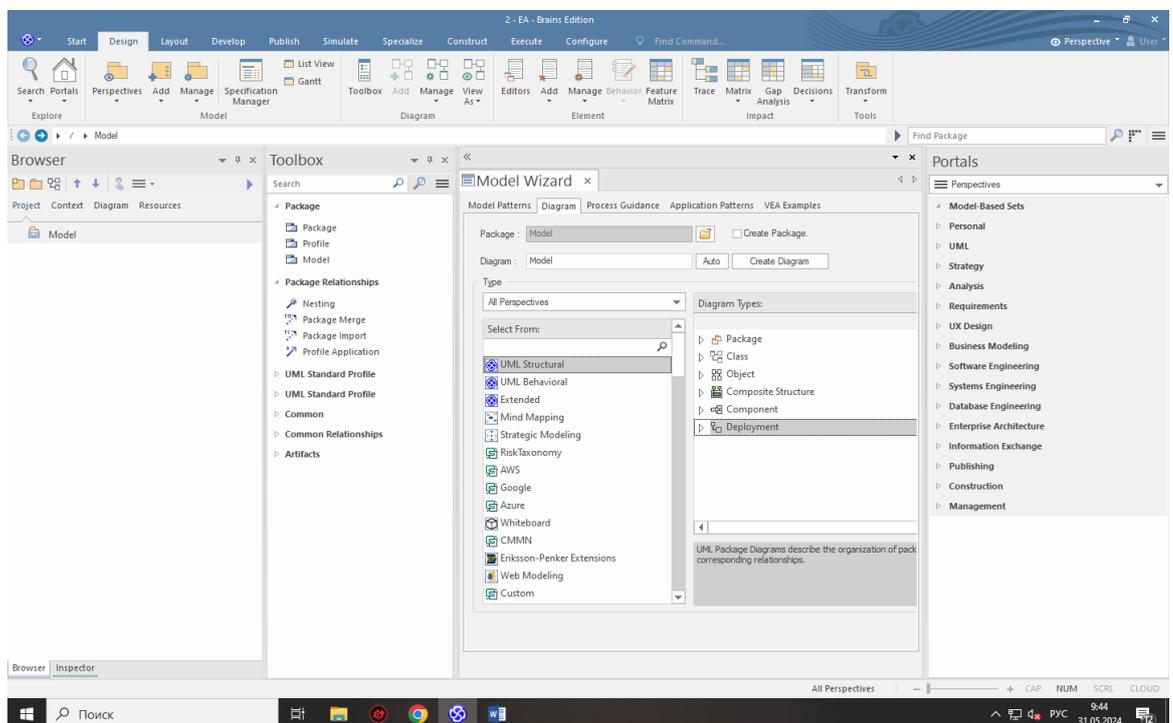
В окне под названием Browser, находящейся в левой части программы, по созданной пользователем модели нажимаем ПКМ. Вылезает список, где необходимо выбрать «Add a Model using Wizard» (Добавить модель с помощью мастера).



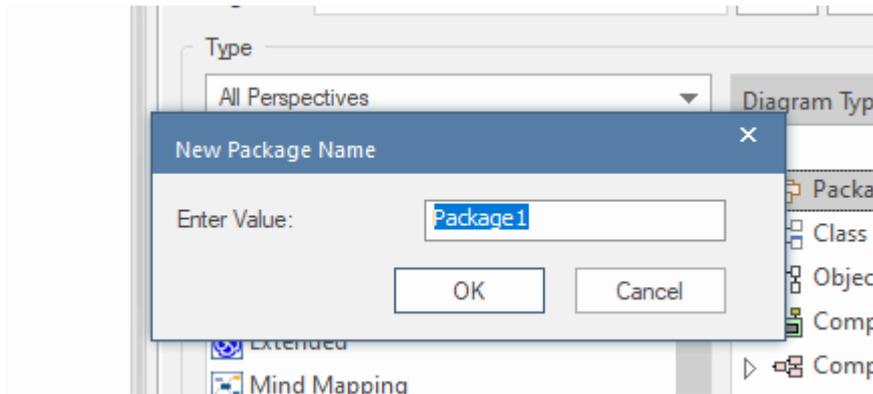
Далее откроется окно «Model Wizard» (Мастер моделей), где необходимо перейти во вкладку Diagram.



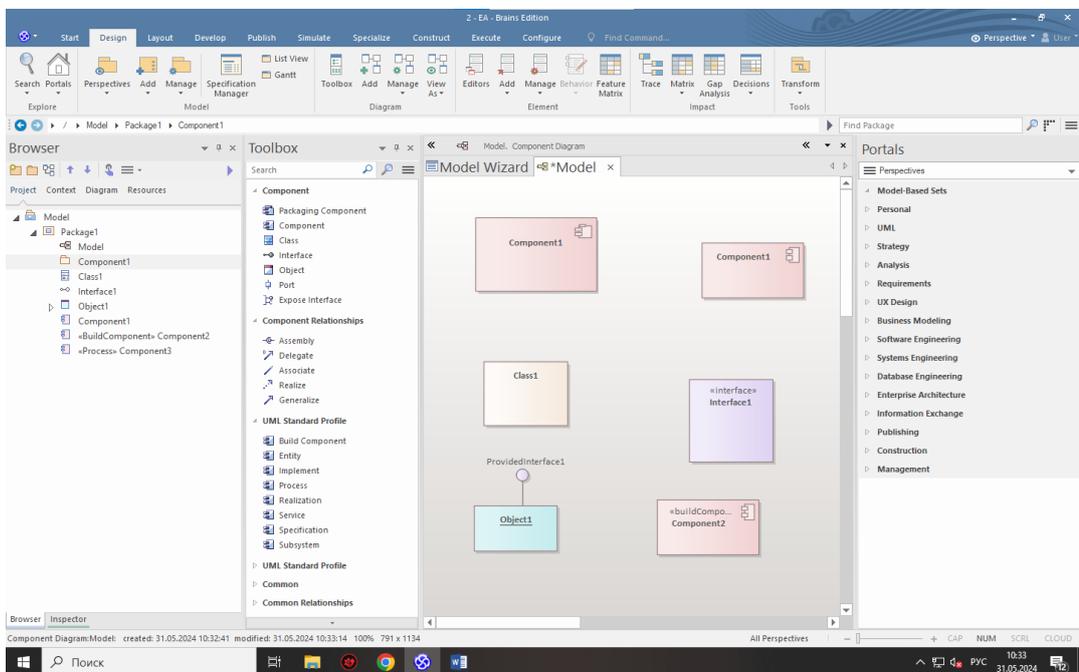
В левом окне списков стандартов необходимо выбрать «UML Structural», после чего, указать, какой тип диаграммы нужно создать (в данном случае, компонентов). Также для создания среды диаграммы необходимо нажать на поле для галочки «Create Package». После этого нажимаем кнопку «Create Diagram».



После появления окна «New Package Name», в поле ввода необходимо ввести название среды для работы с диаграммой. После этого нажимаем кнопку «ОК».



В левом окне «Browser» необходимо выбрать созданный пакет, после чего в среднем окне появится вкладка с названием созданной пользователем модели. В панели с названием «Toolbox» необходимо выбрать варианты компонентов, которые необходимо спроектировать.



Значки панели инструментов элемента диаграммы компонентов

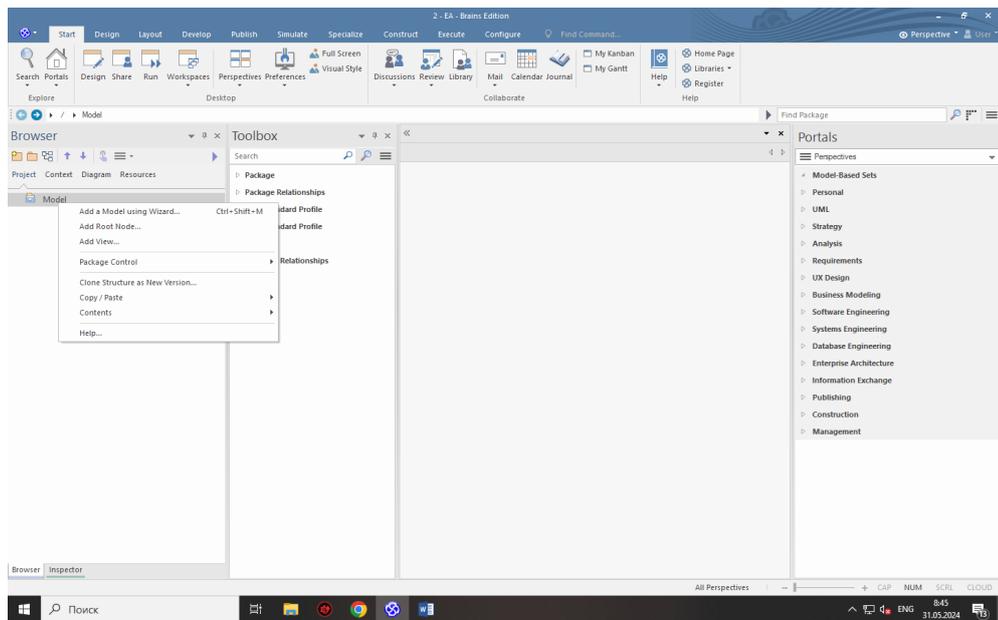
Иконка	Описание
 Packaging Component	Компонент упаковки — это элемент, который очень похож на компонент на диаграмме, но ведет себя как пакет в окне браузера .
 Component	Компонент — это модульная часть системы, поведение которой определяется предоставляемыми и необходимыми интерфейсами.
 Class	Класс — это представление типа объекта, которое отражает структуру и поведение таких объектов внутри системы.
 Interface	Интерфейс — это спецификация поведения (или контракта), которую разработчики соглашаются соблюдать.
 Object	Объект — это конкретный экземпляр класса во время выполнения.
 Port	Порты определяют взаимодействие между классификатором и его средой.
 Expose Interface	Элемент Expose Interface — это графический метод изображения требуемых или предоставляемых интерфейсов Компонента, Класса или Части на диаграмме компонента или составной структуры .

Значки панели инструментов соединителя диаграммы компонентов

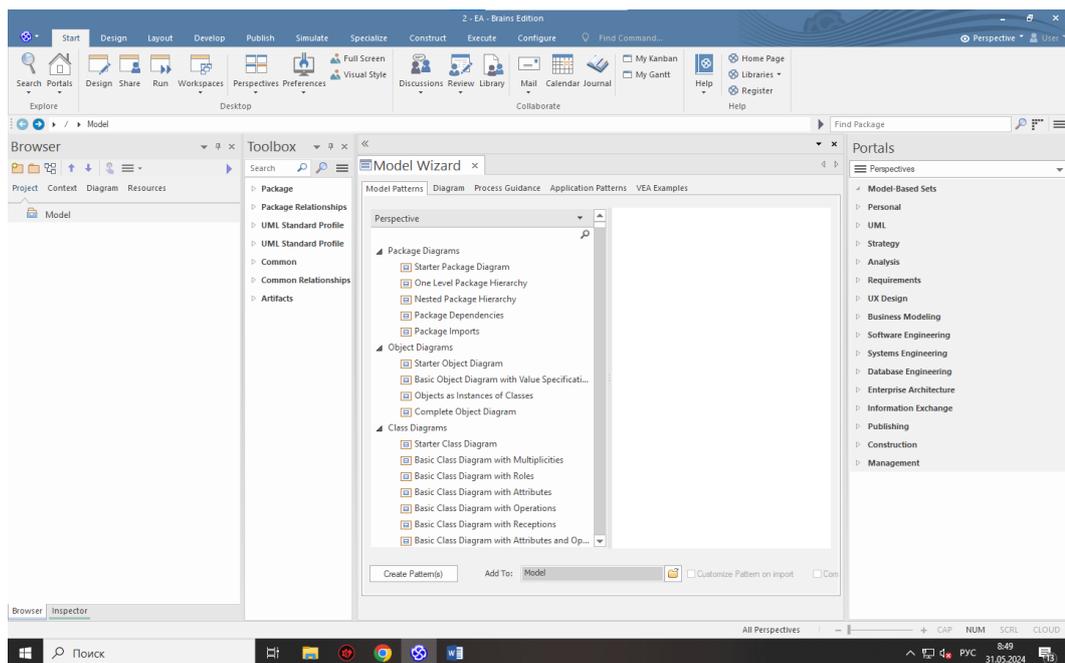
Иконка	Описание
 Assembly	Соединитель сборки соединяет требуемый интерфейс компонента (Компонент1) с предоставленным интерфейсом другого компонента (Компонент2), обычно на диаграмме компонентов.
 Delegate	Соединитель делегата определяет внутреннюю сборку внешних портов и интерфейсов компонента на диаграмме компонентов.
 Associate	Ассоциация подразумевает, что два элемента модели имеют связь, обычно реализуемую как переменную экземпляра в одном или обоих классах.
 Realize	Исходный объект реализует или реализует свой целевой объект. Соединители Realize используются в диаграммах вариантов использования, компонентов или требований для обеспечения прослеживаемости и полноты модели.
 Generalize	Обобщение используется для обозначения наследования.

Диаграмма вариантов использования (use case)

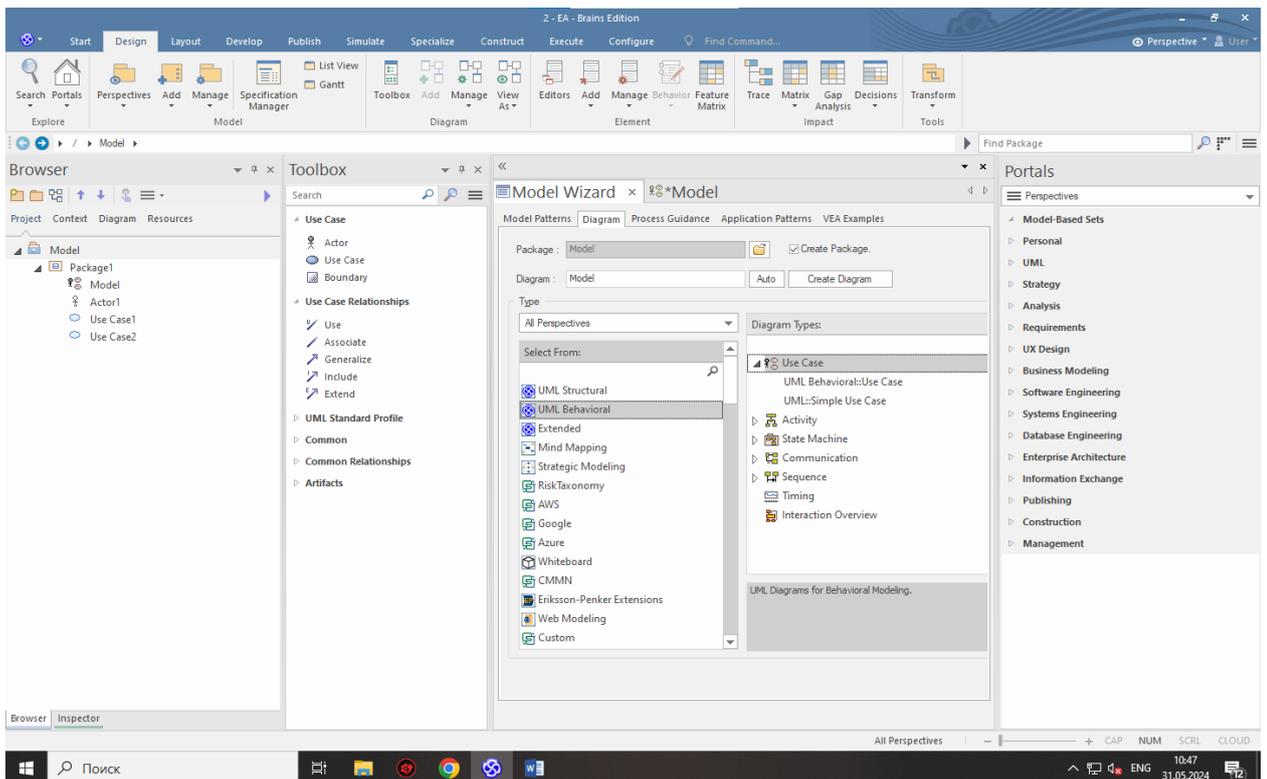
В окне под названием Browser, находящейся в левой части программы, по созданной пользователем модели нажимаем ПКМ. Вылезает список, где необходимо выбрать «Add a Model using Wizard» (Добавить модель с помощью мастера).



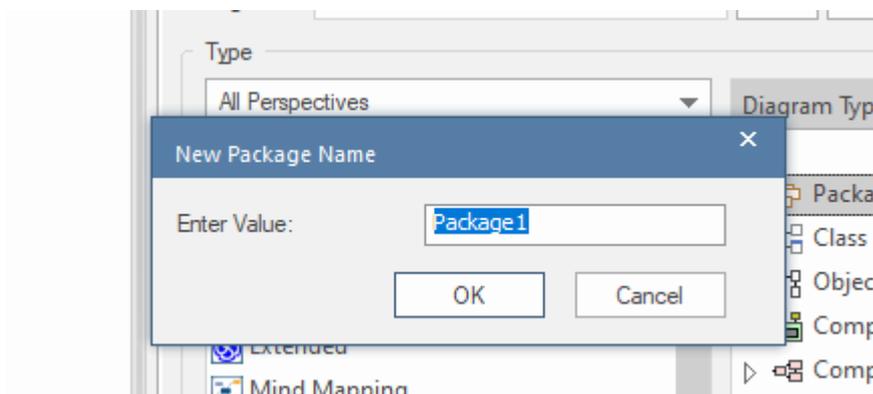
Далее откроется окно «Model Wizard» (Мастер моделей), где необходимо перейти во вкладку Diagram.



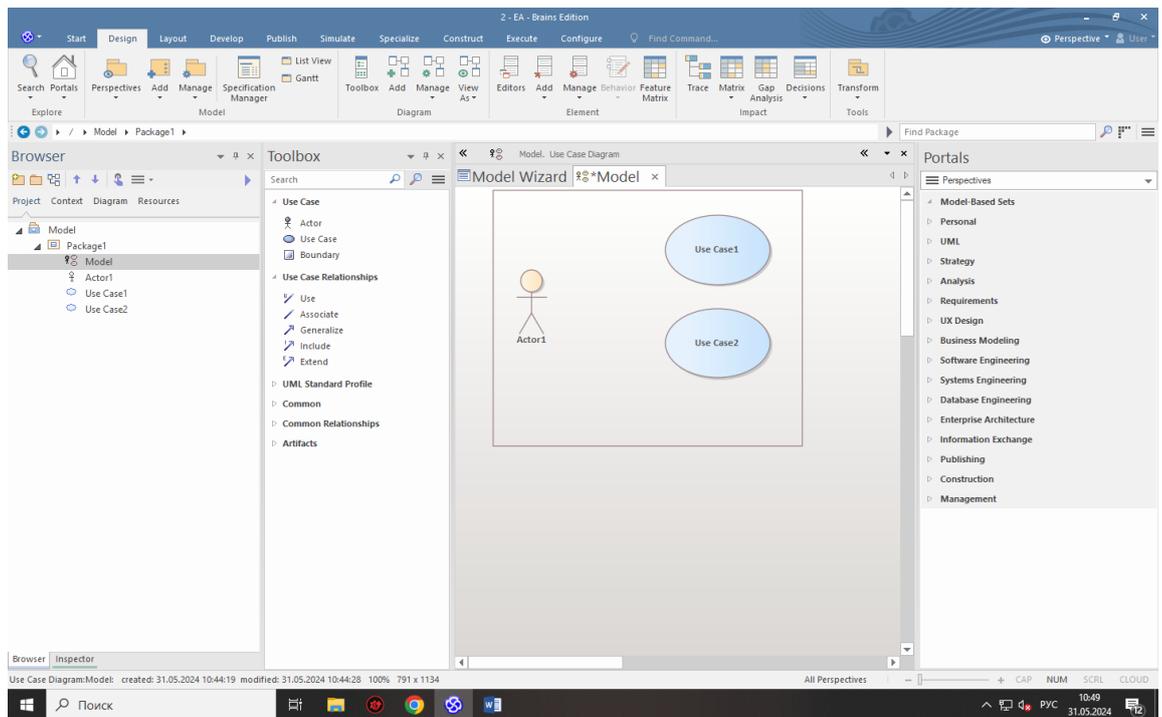
В левом окне списков стандартов необходимо выбрать «UML Behavioral», после чего, указать, какой тип диаграммы нужно создать (в данном случае, вариантов использования). Также для создания среды диаграммы необходимо нажать на поле для галочки «Create Package». После этого нажимаем кнопку «Create Diagram».



После появления окна «New Package Name», в поле ввода необходимо ввести название среды для работы с диаграммой. После этого нажимаем кнопку «ОК».



В левом окне «Browser» необходимо выбрать созданный пакет, после чего в среднем окне появится вкладка с названием созданной пользователем модели. В панели с названием «Toolbox» необходимо выбрать варианты использования, которые необходимо спроектировать.



Сценарий использования

Основной аспект моделирования вариантов использования выполняется с помощью функции «Сценарий варианта использования», которая помогает добавить более подробное текстовое представление базового варианта использования.

Type:		Scenario:		
Basic Path		Basic Path		
Step	Action	Uses	Results	State
1	The driver clicks the remote control for keyless entry.			
2	The system validates the signal and unlocks the car doors.			
3	The driver opens the driver's door and sits in the driving seat.			
4	new step...			

Используя конструктор сценариев, на основе этих текстовых деталей можно затем создать диаграммы поведенческой модели, чтобы обеспечить основу для более детального проектирования.

Элементы

Основными элементами, которые могут отображаться на диаграммах вариантов использования, являются:

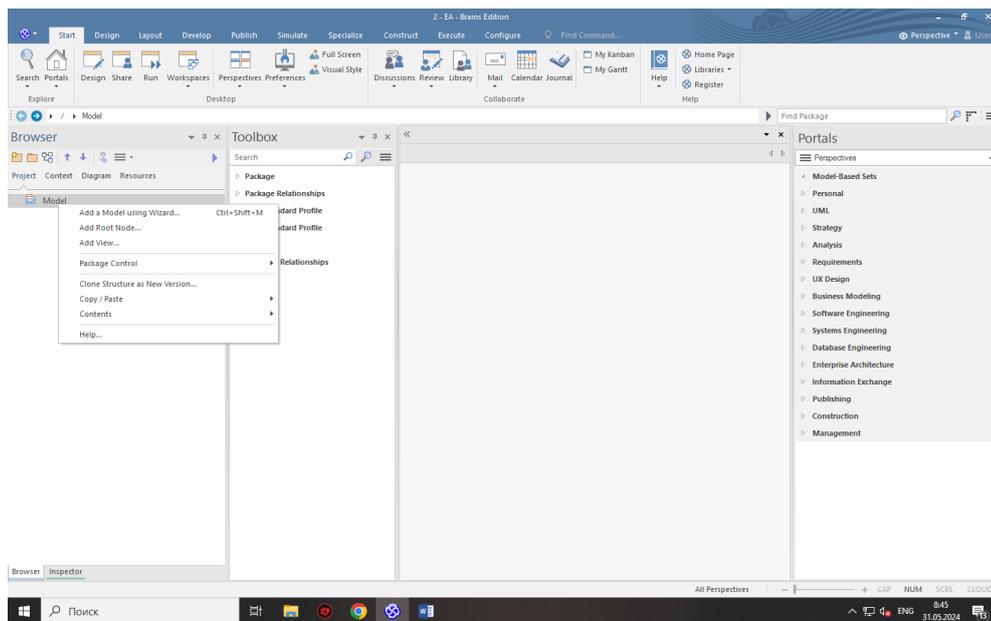
- Актер
- Вариант использования
- Граница

Основные соединители, которые могут отображаться на диаграммах вариантов использования:

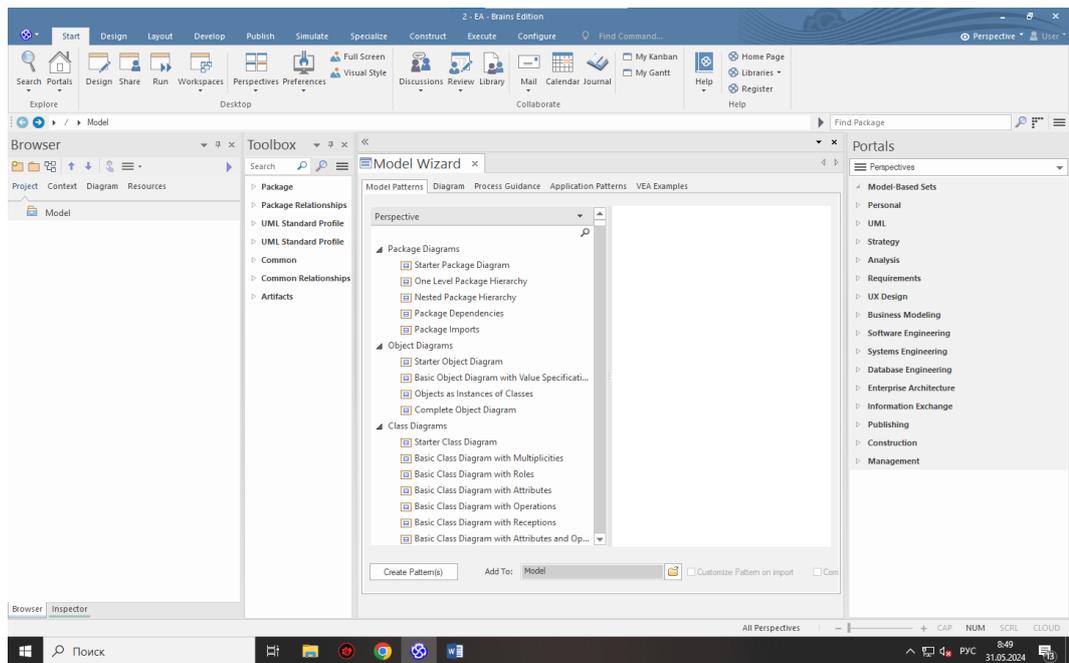
- Путь связи
- Обобщать
- Включать
- Продлевать

Диаграмма деятельности (Activity Diagram)

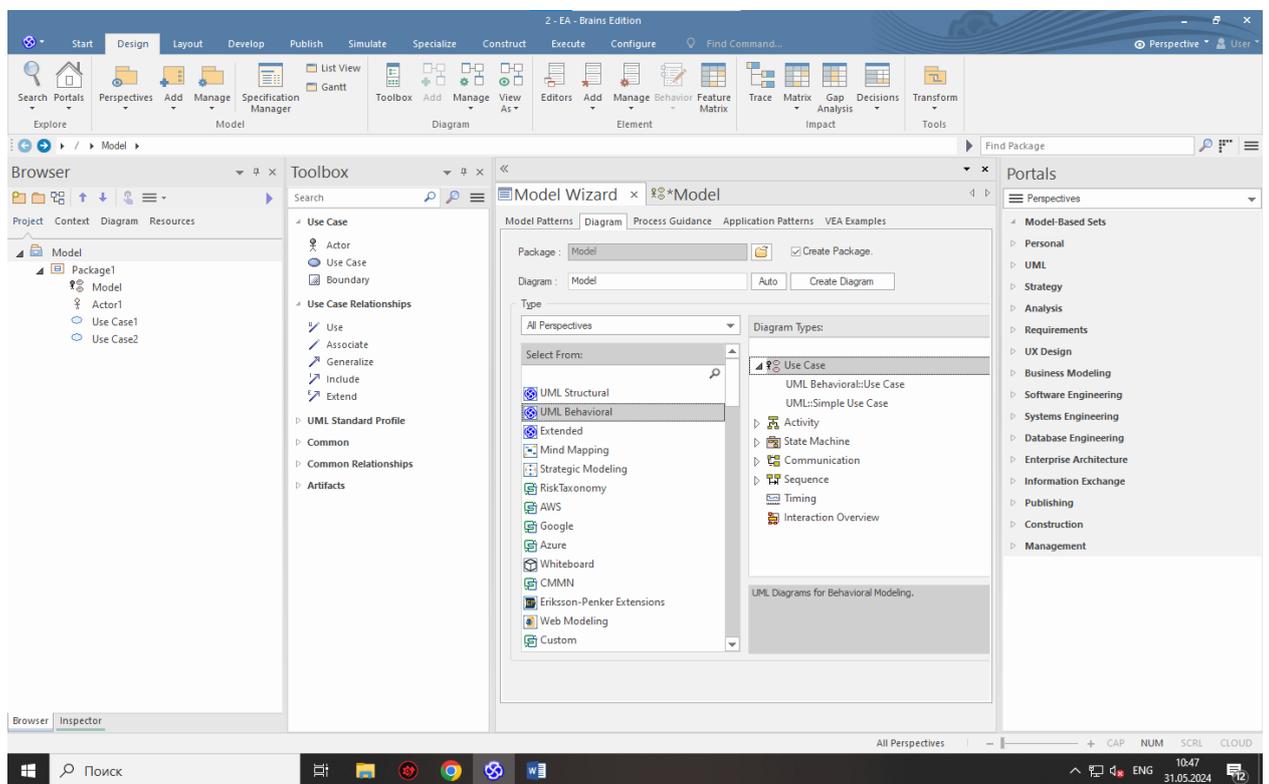
В окне под названием Browser, находящейся в левой части программы, по созданной пользователем модели нажимаем ПКМ. Вылезает список, где необходимо выбрать «Add a Model using Wizard» (Добавить модель с помощью мастера).



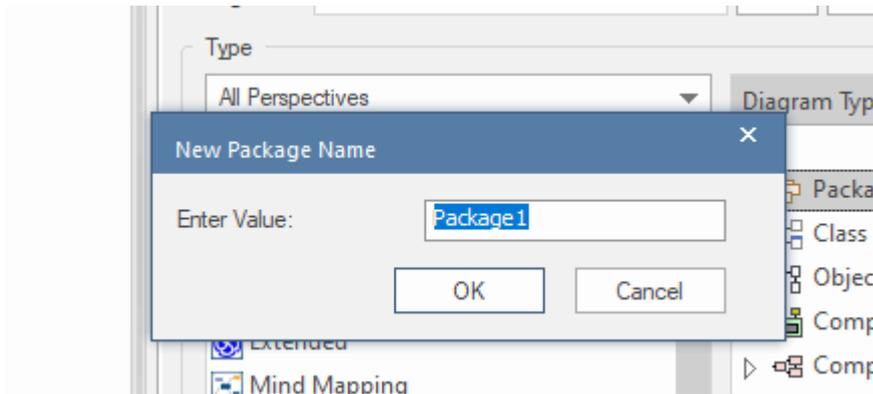
Далее откроется окно «Model Wizard» (Мастер моделей), где необходимо перейти во вкладку Diagram.



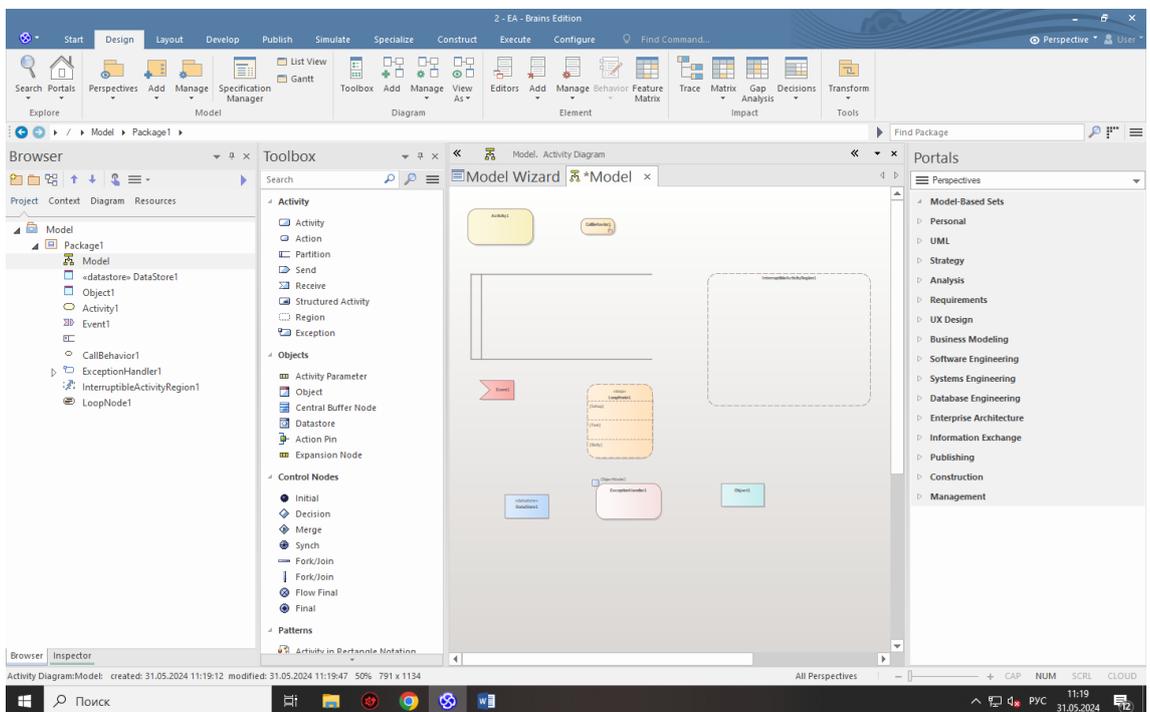
В левом окне списков стандартов необходимо выбрать «UML Behavioral», после чего, указать, какой тип диаграммы нужно создать (в данном случае, деятельности). Также для создания среды диаграммы необходимо нажать на поле для галочки «Create Package». После этого нажимаем кнопку «Create Diagram».



После появления окна «New Package Name», в поле ввода необходимо ввести название среды для работы с диаграммой. После этого нажимаем кнопку «OK».



В левом окне «Browser» необходимо выбрать созданный пакет, после чего в среднем окне появится вкладка с названием созданной пользователем модели. В панели с названием «Toolbox» необходимо выбрать объекты, которые необходимо спроектировать.



Значки панели инструментов элемента диаграммы деятельности

Иконки	Описание
 Activity	Элемент Activity организует и определяет участие подчиненных действий, таких как поддействия или действия, для отражения управления и потока данных процесса.
 Action	Элемент «Действие» описывает базовый процесс или преобразование, происходящее внутри системы, и является базовой функциональной единицей диаграммы действий.
 Partition	Элемент Partition используется для логической организации элементов Activity.
 Send	Элемент «Отправить» отображает действие по отправке сигнала на диаграмме действий.
 Receive	Элемент приема определяет принятие или получение запроса на диаграмме действий.
 Structured Activity	Структурированная деятельность — это узел активности, который может иметь подчиненные узлы в качестве независимой группы действий.
 Region	Область расширения окружает процесс, который будет многократно применяться к входящим данным, по одному разу для каждого элемента входной коллекции.
 Exception	Элемент Exception Handler определяет группу операций, которые необходимо выполнить при возникновении исключения.
 Activity Parameter	Узел параметров действия принимает входные данные для действия или обеспечивает выходные данные действия.
 Object	Объект — это конкретный экземпляр класса во время выполнения.
 Central Buffer Node	Центральный буферный узел — это объектный узел для управления потоками из нескольких источников и пунктов назначения, представленный на диаграмме действий.
 Datastore	Хранилище данных определяет постоянно хранящиеся данные.
 Action Pin	Пин действия используется для определения значений данных, передаваемых из действия и в него. Входной контакт предоставляет значения для действия, тогда как выходной контакт содержит результаты этого действия.
 Expansion Node	Узел расширения — это сокращенное обозначение, обозначающее, что действие/деятельность состоит из области расширения.
 Initial	Элемент Initial используется для определения начала потока при вызове действия.
 Decision	На диаграмме действий или диаграмме обзора взаимодействия Решение указывает точку условной прогрессии: если условие имеет значение True , то обработка продолжается в одном направлении; если нет, то другой.

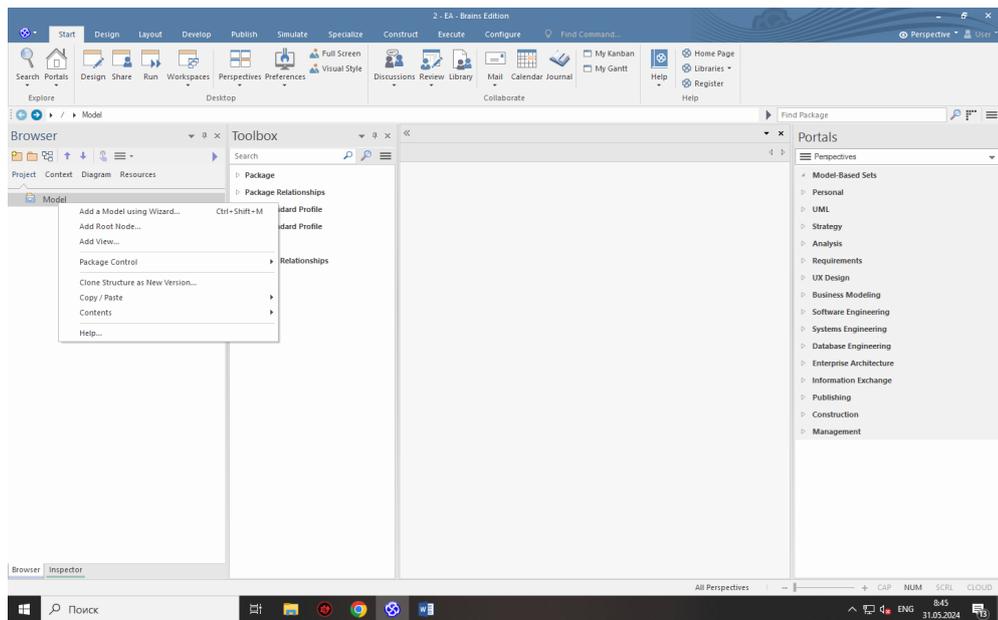
 Merge	Узел слияния объединяет ряд альтернативных путей потока на диаграммах действий, анализа и обзора взаимодействия.
 Synch	Состояние синхронизации полезно для указания того, что одновременные пути StateMachine синхронизированы. Он используется для разделения и объединения периодов параллельной обработки.
 Fork/Join	Элемент Fork/Join можно использовать для: 1) Разделить один поток на несколько параллельных потоков. 2) Присоединиться к нескольким параллельным потокам или 3) Оба объединяют и разветвляют несколько входящих потоков на несколько исходящих потоков.
 Fork/Join	
 Flow Final	Элемент Flow Final изображает выход из системы, в отличие от Activity Final, который представляет собой завершение действия.
 Final	Элемент Activity Final указывает на завершение действия; при достижении финала все выполнение на диаграмме активности прерывается.

Значки панели инструментов соединителя диаграммы компонентов

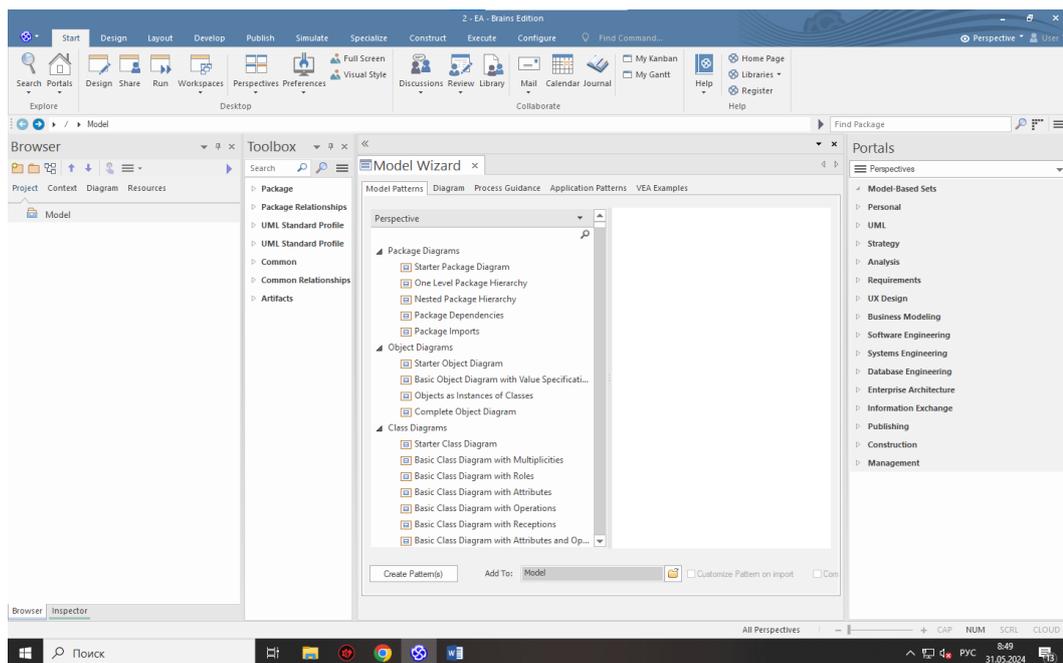
Иконка	Описание
 Control Flow	Поток управления соединяет два узла на диаграмме действий, моделируя активный переход.
 Object Flow	Поток объектов соединяет два элемента, через которые проходят определенные данные, моделируя активный переход.
 Interrupt Flow	Поток прерываний определяет две концепции UML: соединители для обработчика исключений и области прерываемой активности.

Диаграмма состояний (State machine)

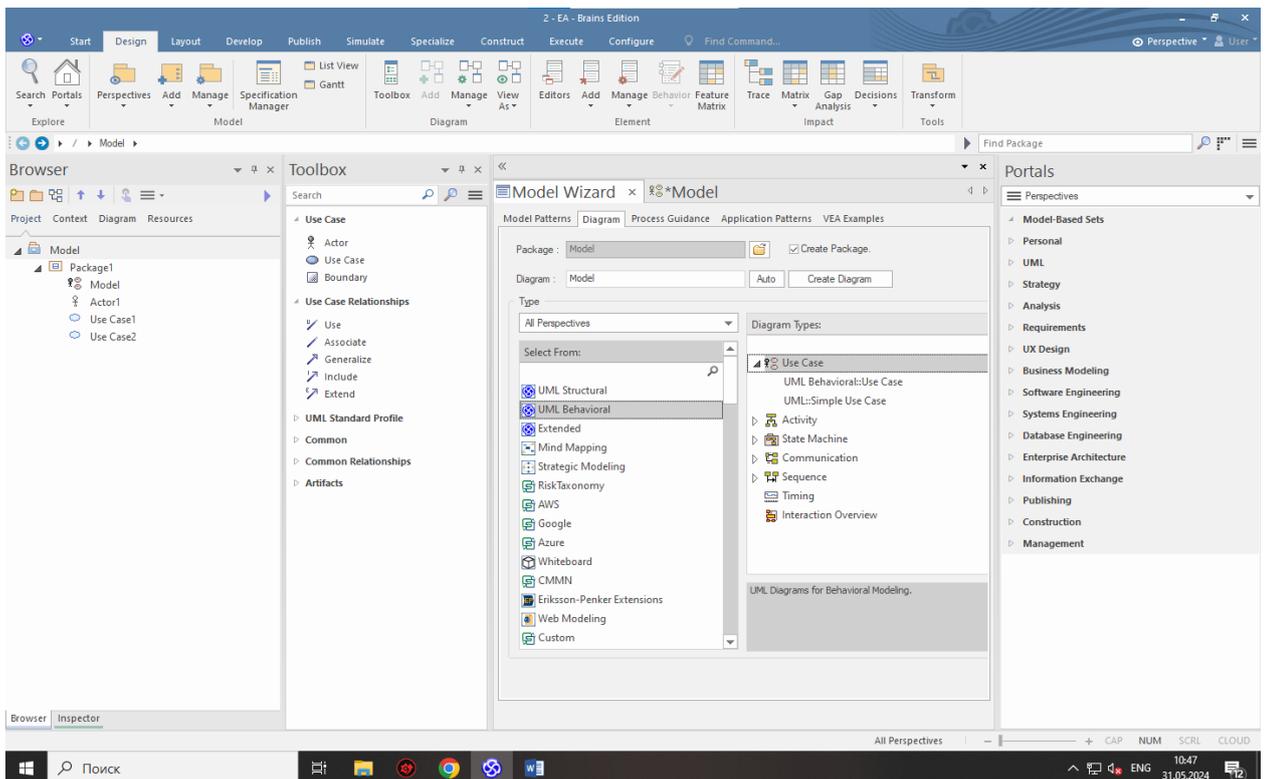
В окне под названием Browser, находящейся в левой части программы, по созданной пользователем модели нажимаем ПКМ. Вылезает список, где необходимо выбрать «Add a Model using Wizard» (Добавить модель с помощью мастера).



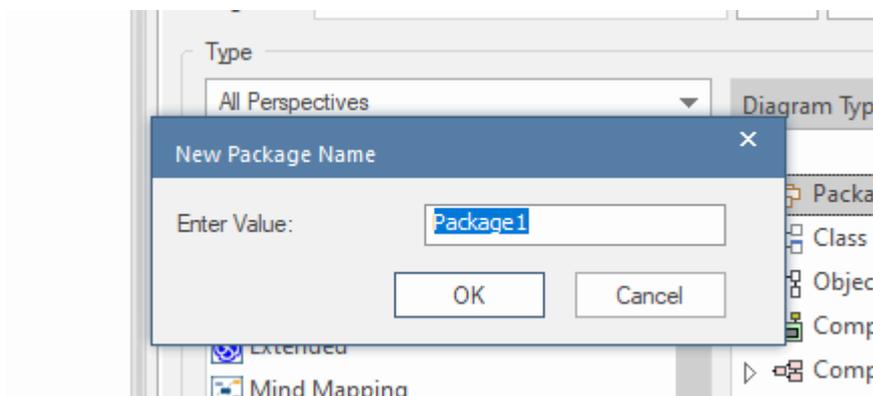
Далее откроется окно «Model Wizard» (Мастер моделей), где необходимо перейти во вкладку Diagram.



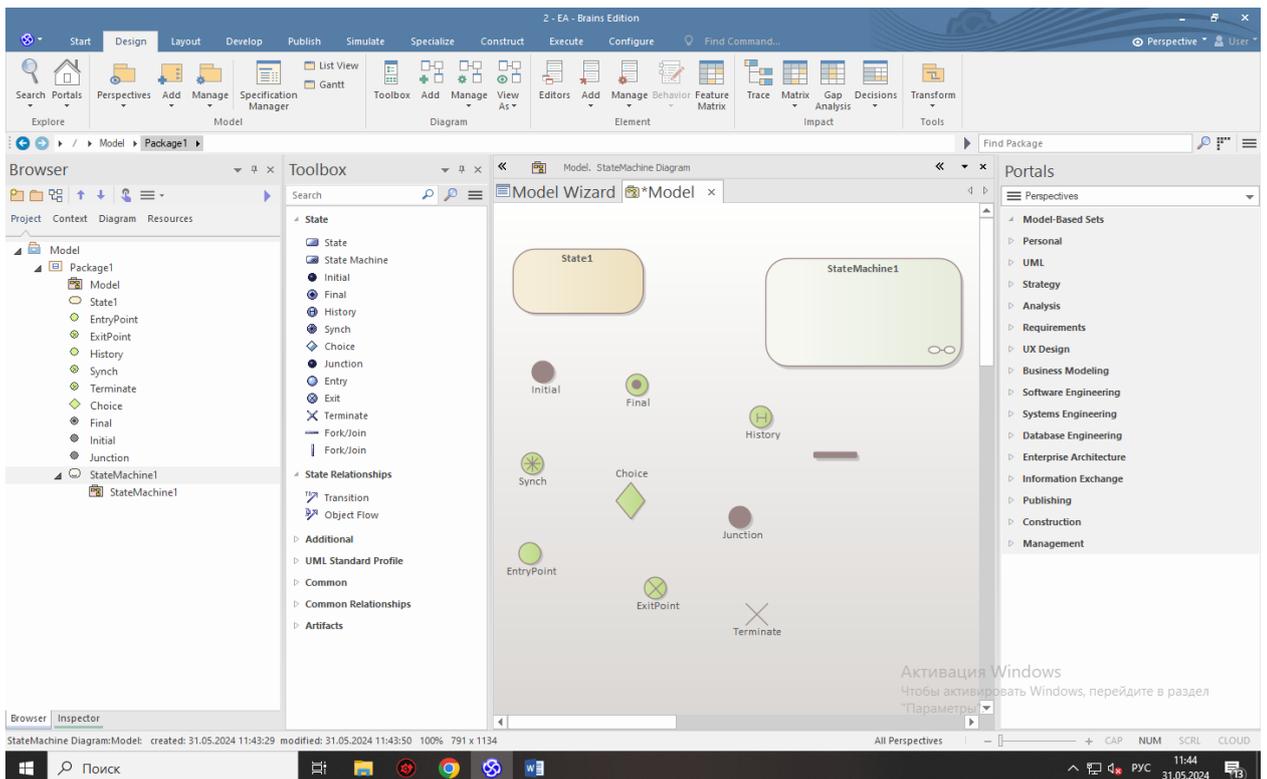
В левом окне списков стандартов необходимо выбрать «UML Behavioral», после чего, указать, какой тип диаграммы нужно создать (в данном случае, состояний). Также для создания среды диаграммы необходимо нажать на поле для галочки «Create Package». После этого нажимаем кнопку «Create Diagram».



После появления окна «New Package Name», в поле ввода необходимо ввести название среды для работы с диаграммой. После этого нажимаем кнопку «ОК».



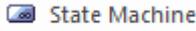
В левом окне «Browser» необходимо выбрать созданный пакет, после чего в среднем окне появится вкладка с названием созданной пользователем модели. В панели с названием «Toolbox» необходимо выбрать состояния, которые необходимо спроектировать.

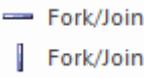


«ProtocolStateMachines используются для выражения протоколов использования. ProtocolStateMachines выражают юридические последовательности возникновения событий, которым должно соответствовать поведение связанного BehavedClassifier. Нотация StateMachine — это удобный способ определить порядок вызовов поведенческих функций классификатора. ProtocolStateMachines могут быть связаны с классификаторами, интерфейсами и портами».

Чтобы создать ProtocolStateMachine, создайте элемент StateMachine и откройте окно свойств для этого элемента. Выберите вкладку «Поведение» и установите флажок «Протокол StateMachine». Элемент на диаграмме теперь имеет слово <<протокол>> над именем элемента.

Значки панели инструментов элемента диаграммы состояний

Иконка	Описание
 State	Состояние представляет собой ситуацию, в которой выполняется некоторое инвариантное условие; это состояние может быть статическим (ожидание события) или динамическим (выполнение набора действий).
 State Machine	Элемент StateMachine — это контейнер для групп связанных элементов State.
 Initial	Элемент Initial представляет псевдосостояние, используемое для обозначения состояния по умолчанию составного состояния; в каждом регионе составного состояния может быть одна начальная вершина.

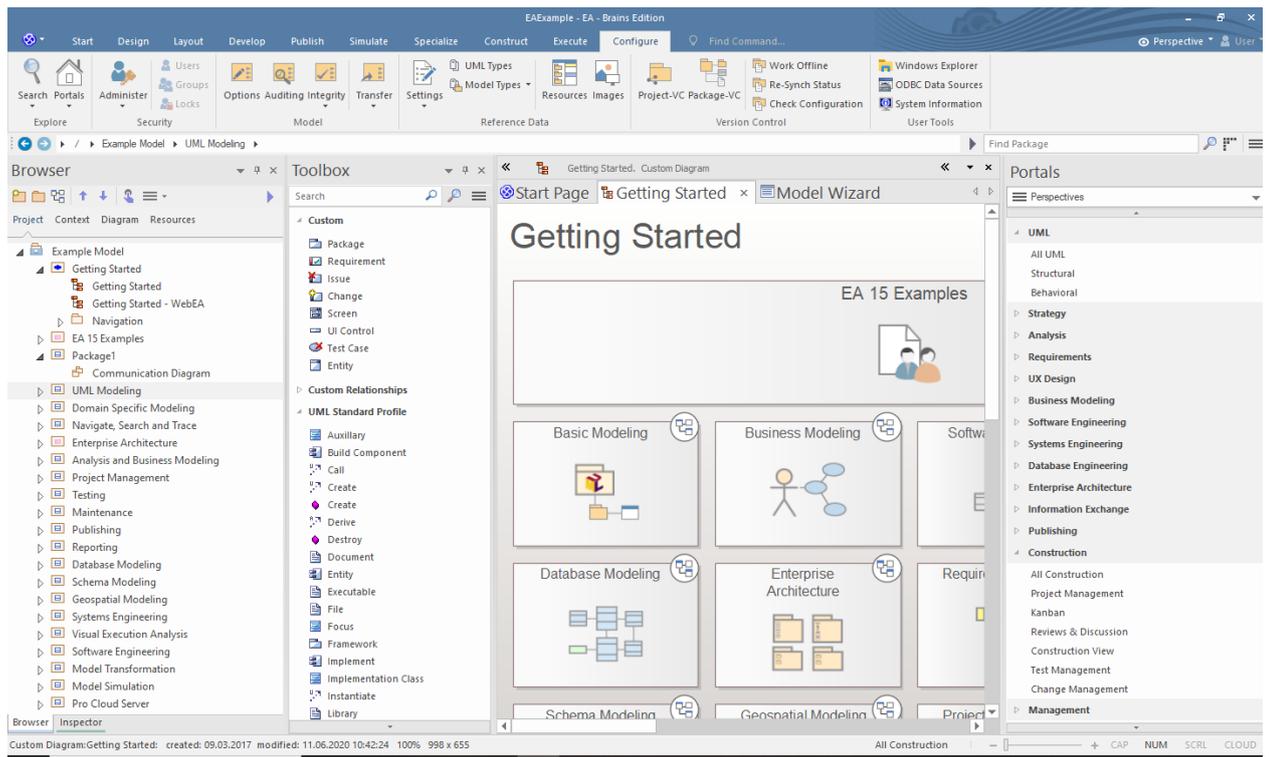
 Final	Элемент Activity Final указывает на завершение действия; при достижении финала все выполнение на диаграмме активности прерывается.
 History -	В UML определены два типа псевдосостояния истории: неглубокая и глубокая история.
 Synch	Состояние синхронизации полезно для указания того, что одновременные пути StateMachine синхронизированы. Они используются для разделения и объединения периодов параллельной обработки.
 Choice	Псевдосостояние «Выбор» используется для составления сложных переходных путей, где исходящий путь перехода определяется динамическими условиями времени выполнения.
 Junction	Псевдосостояния соединений используются для проектирования сложных переходных путей в диаграммах StateMachine. Соединение можно использовать для объединения или объединения нескольких путей в общий путь перехода.
 Entry	Псевдосостояния точки входа используются для определения начала StateMachine. Для каждого региона существует точка входа, управляющая начальной конфигурацией одновременного состояния
 Exit	Точки выхода используются в элементах StateMachine и диаграммах StateMachine для обозначения точки выхода из машины и перехода, вызывающего эту точку выхода.
 Terminate	Псевдосостояние Terminate указывает, что при входе в его псевдосостояние выполнение StateMachine заканчивается.
 Fork/Join Fork/Join	Элемент Fork/Join может использоваться для: 1) разделения одного потока на несколько параллельных потоков, 2) объединения нескольких параллельных потоков или 3) объединения и разветвления нескольких входящих потоков на несколько исходящих потоков.

Значки панели инструментов соединителя диаграммы StateMachine

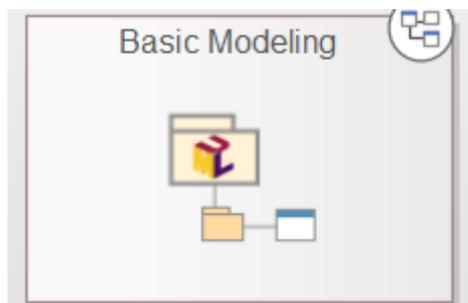
Иконка	Описание
 Transition	Соединитель перехода представляет собой логическое перемещение из одного состояния в другое на диаграмме StateMachine .
 Object Flow	Поток объектов соединяет два элемента, через которые проходят определенные данные, моделируя активный переход.

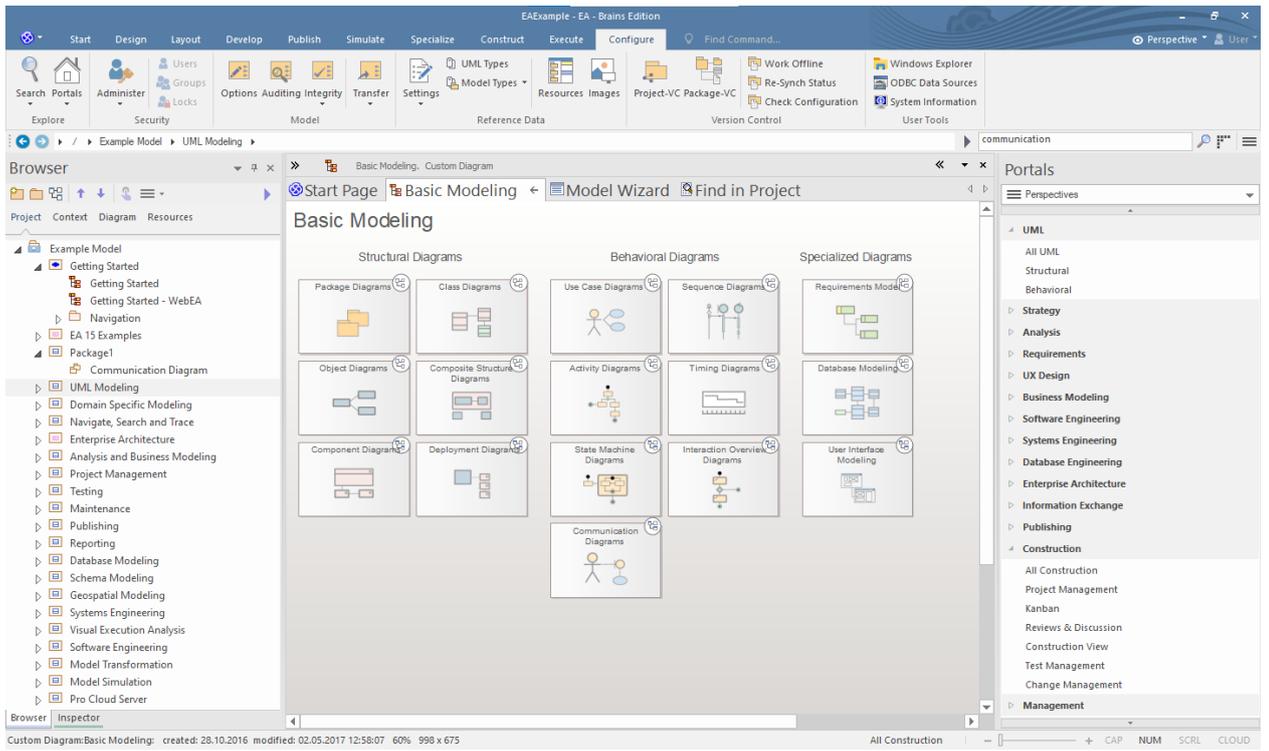
Диаграмма коммуникации (Communication Diagram)

При входе в Enterprise arc. В меню выбрать вкладку Getting Started

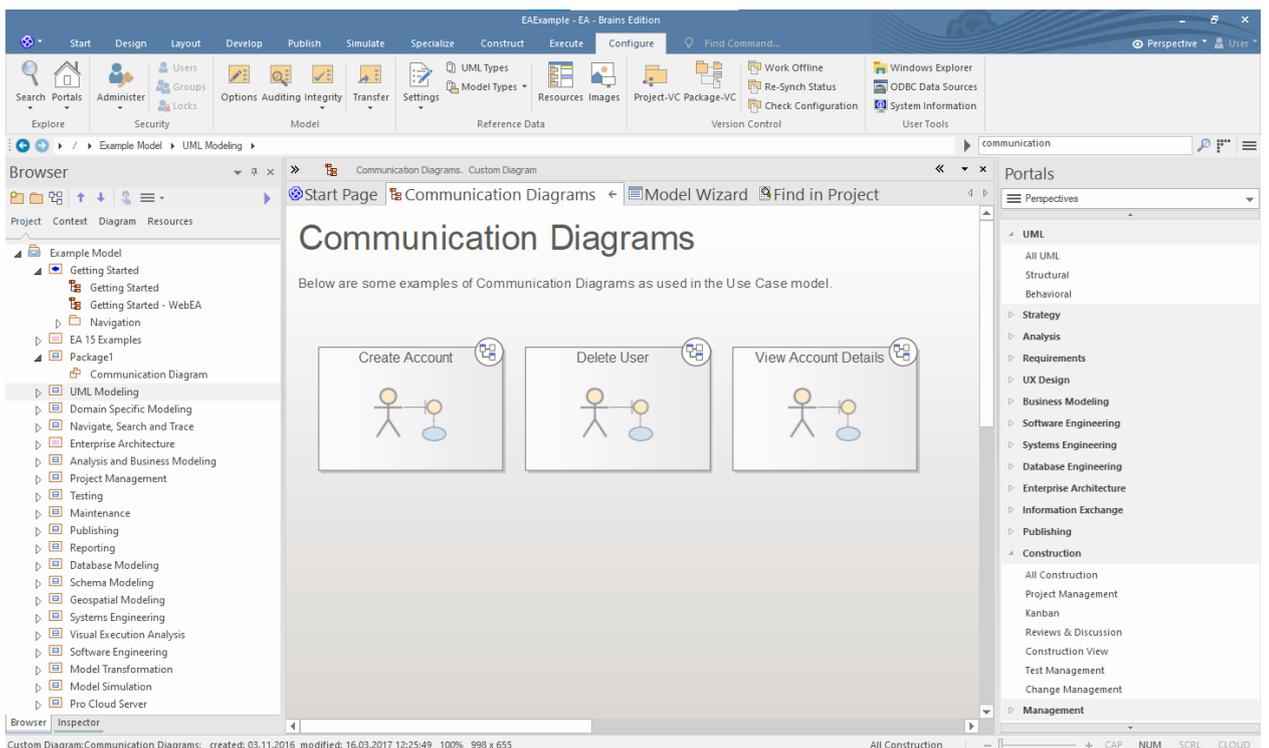
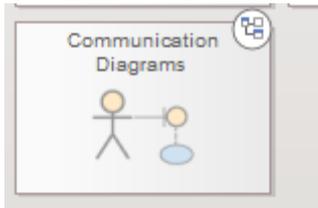


На фото выше выбираем вкладку Basic Modeling (базовое моделирование)





Далее, во вкладке базовых моделей выбираем нужную нам Communication diagrams



Пример диаграммы коммуникации create account:

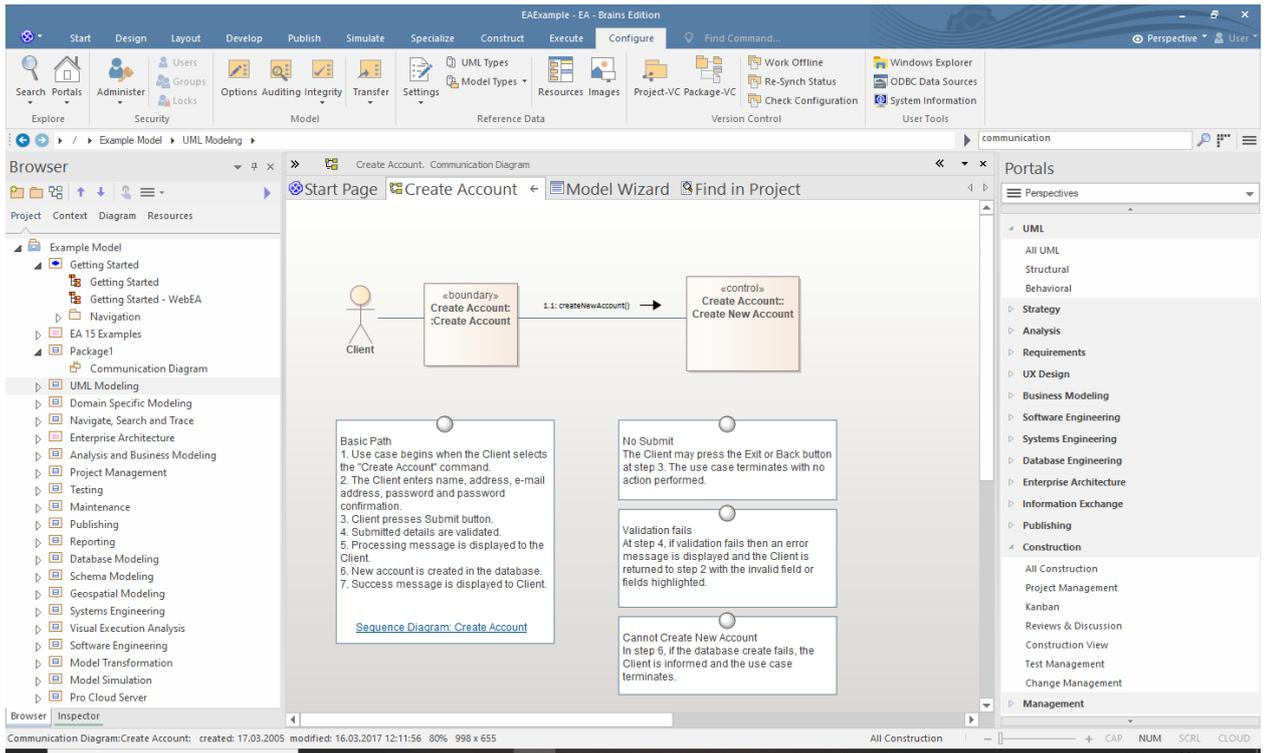
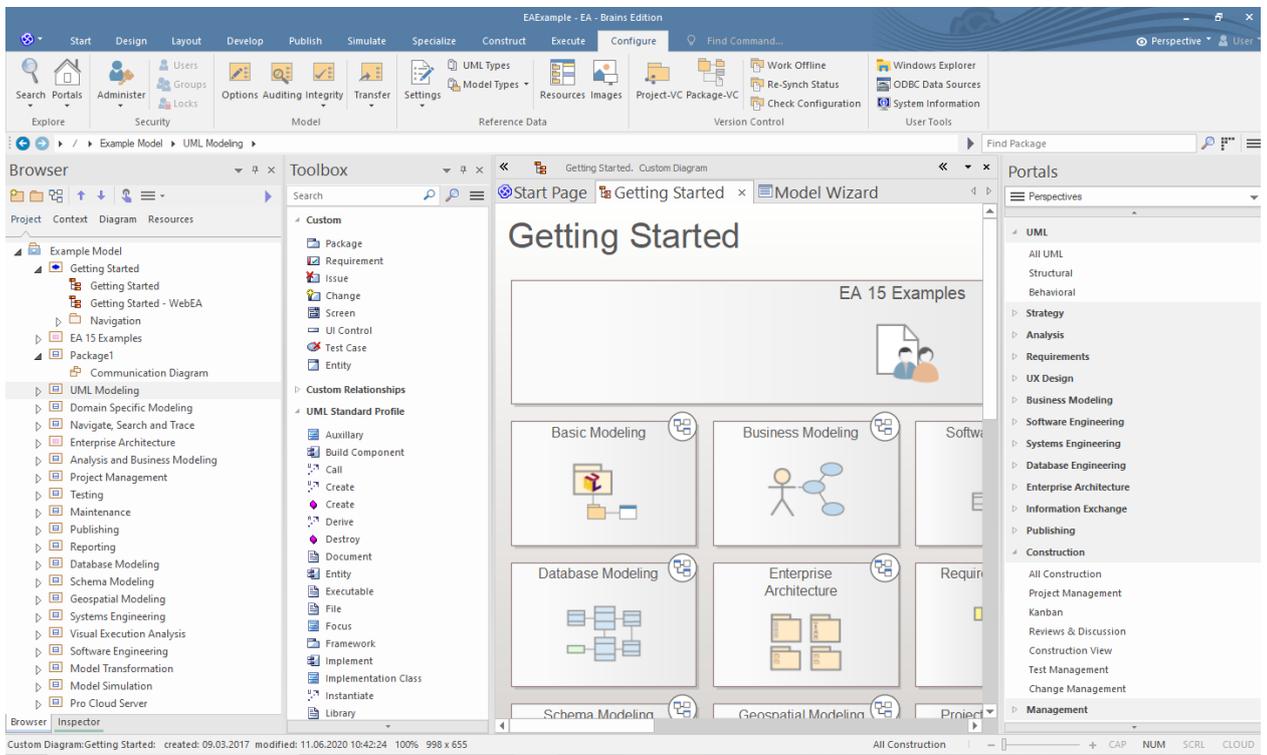
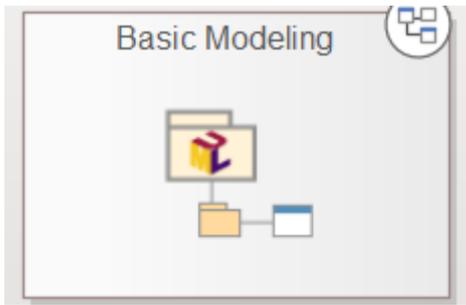


Диаграмма взаимодействия(interaction diagram)

При входе в Enterprise arc. В меню выбрать вкладку Getting Started

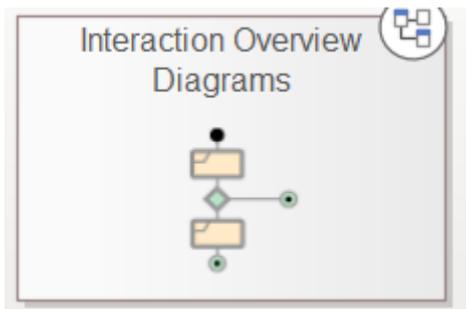


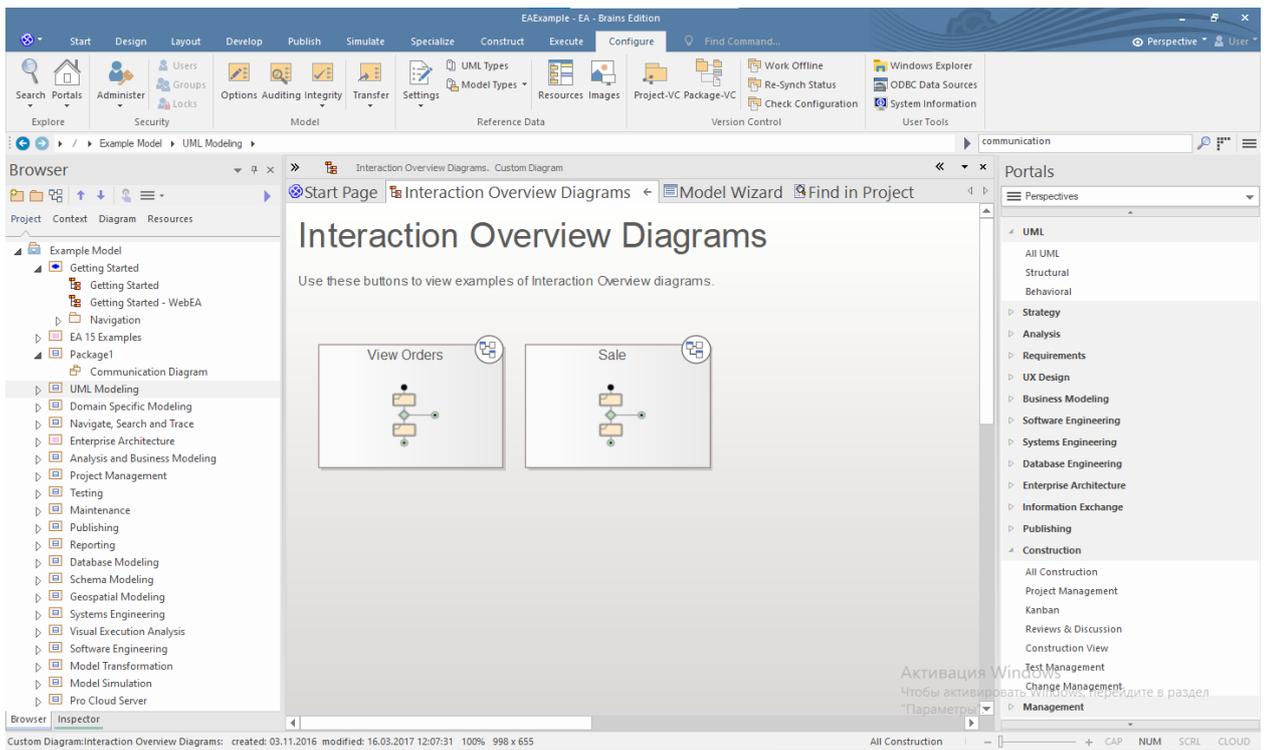
На фото выше выбираем вкладку Basic Modeling (базовое моделирование)



The screenshot shows the EAExample - EA - Brains Edition software interface. The main window displays the "Basic Modeling" diagram, which is a folder icon connected to a document icon. The interface includes a menu bar with options like Start, Design, Layout, Develop, Publish, Simulate, Specialize, Construct, Execute, and Configure. A toolbar contains various icons for search, portals, administration, and model management. The left sidebar shows a project browser with a tree view of the model structure. The right sidebar shows a portals view with a list of perspectives. The status bar at the bottom indicates the diagram was created on 28.10.2016 and modified on 02.05.2017.

Далее, во вкладке базовых моделей выбираем нужную нам interaction Overview diagrams





Пример view orders:

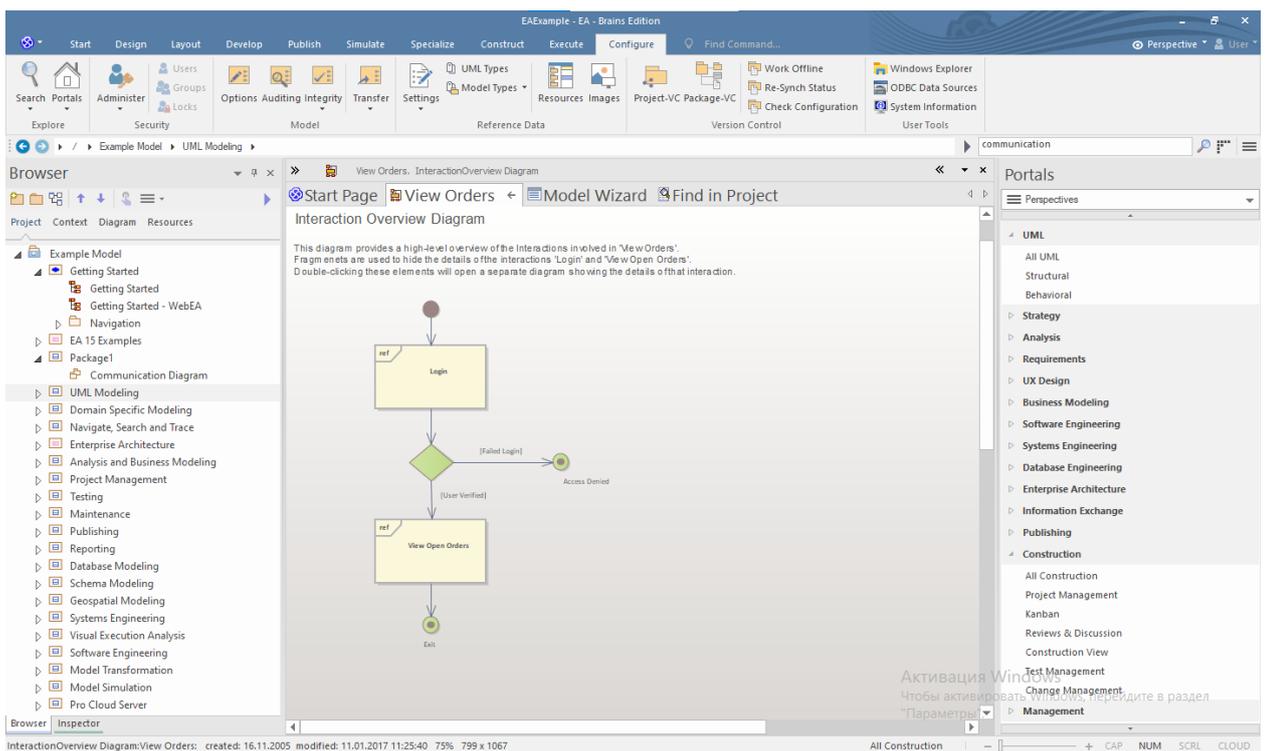
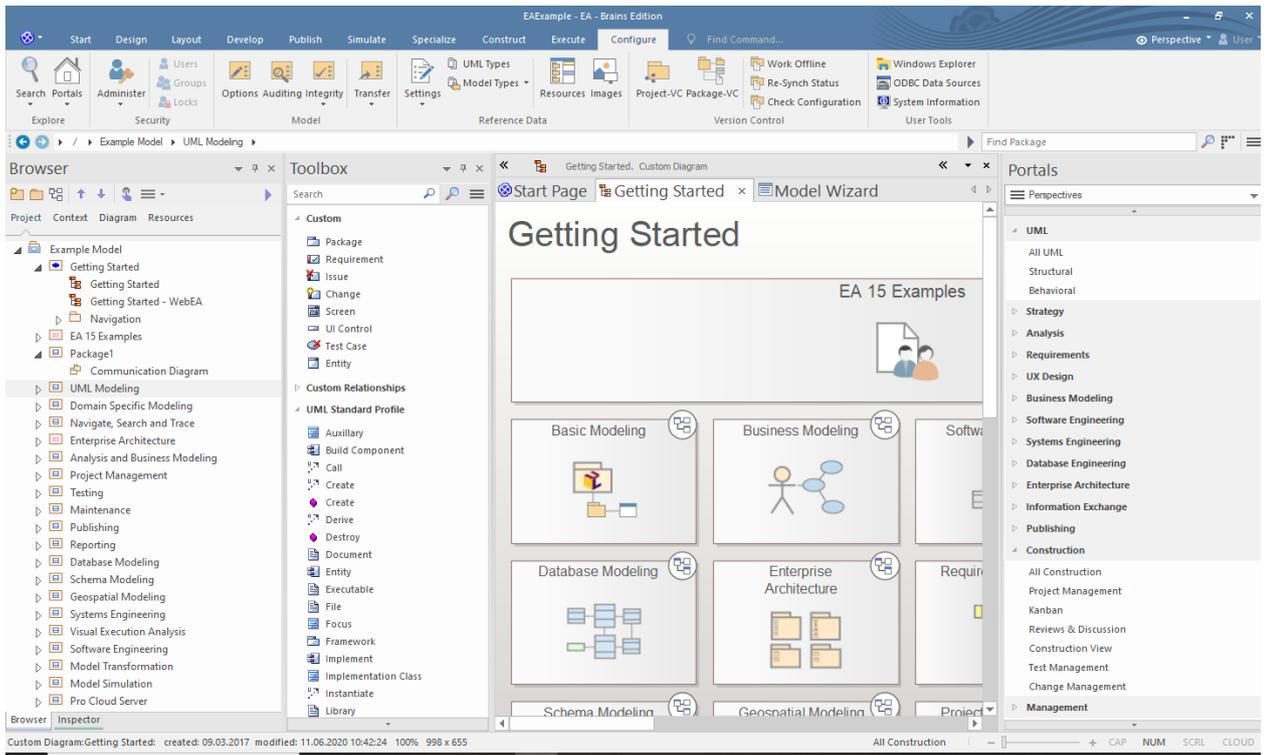
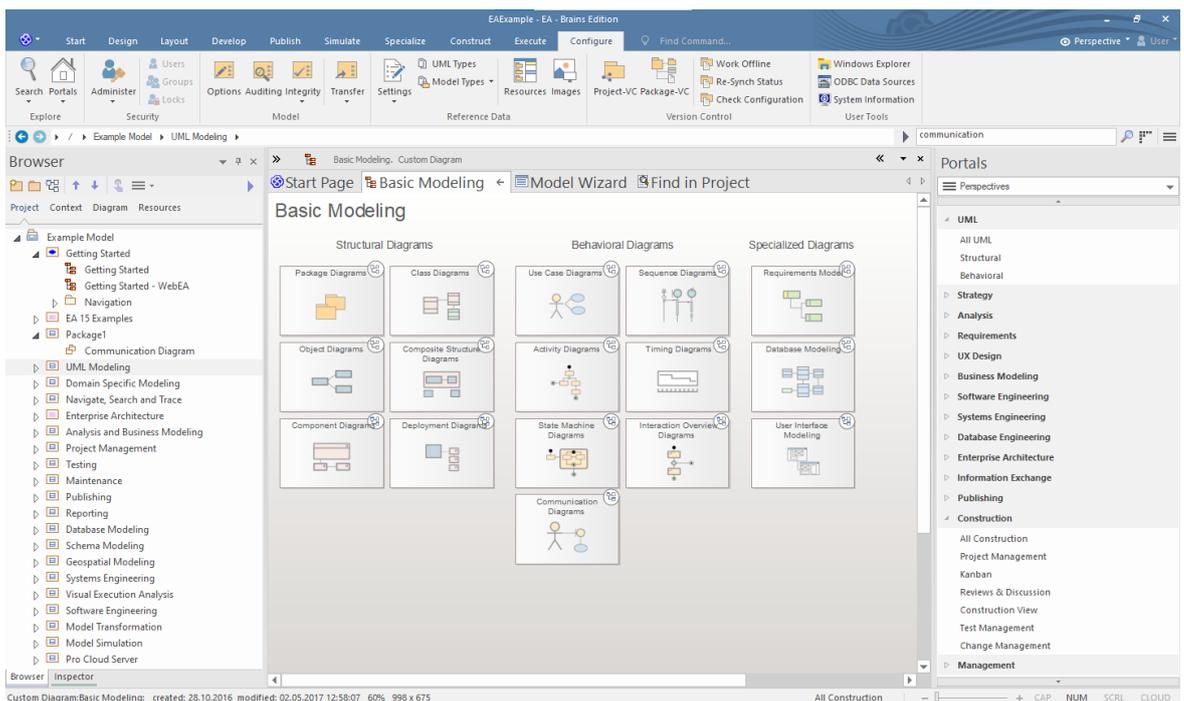
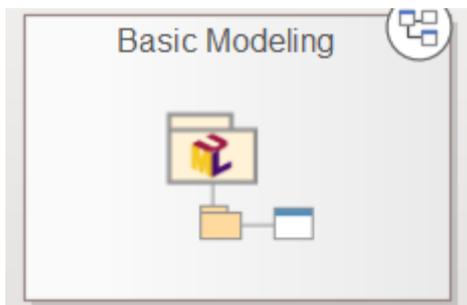


Диаграмма последовательности (sequence diagram)

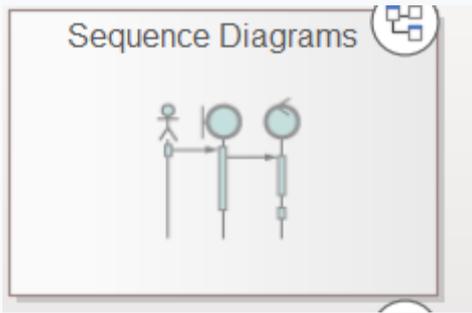
При входе в Enterprise arc. В меню выбрать вкладку Getting Started



На фото выше выбираем вкладку Basic Modeling (базовое моделирование)



Далее, во вкладке базовых моделей выбираем нужную нам sequence diagrams



A screenshot of the EAExample - EA - Brains Edition software interface. The main window displays a slide titled "Sequence Diagrams" with the text "Below are some examples of Sequence Diagrams:" and five small thumbnail images of sequence diagrams. The interface includes a menu bar (Start, Design, Layout, Develop, Publish, Simulate, Specialize, Construct, Execute, Configure), a toolbar with various icons, and a left-hand browser pane showing a project tree. A right-hand pane shows a list of perspectives. The status bar at the bottom indicates "Custom Diagram: Sequence Diagrams: created: 03.11.2016 modified: 16.03.2017 11:27:09 100% 998 x 655".

Пример view history:

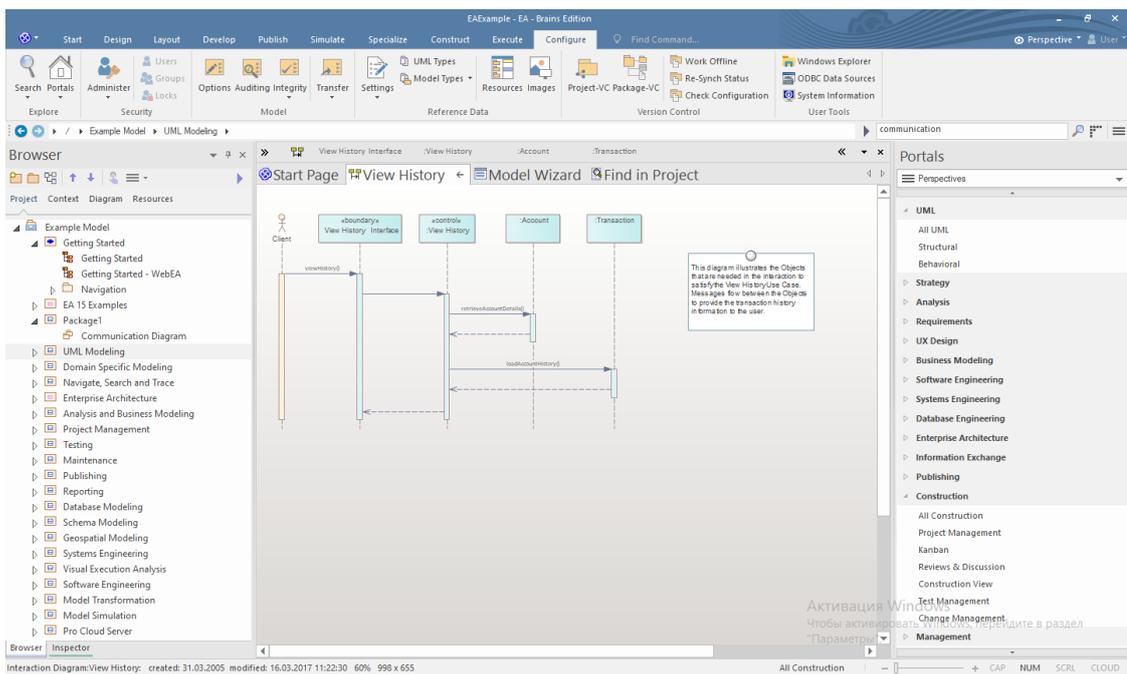
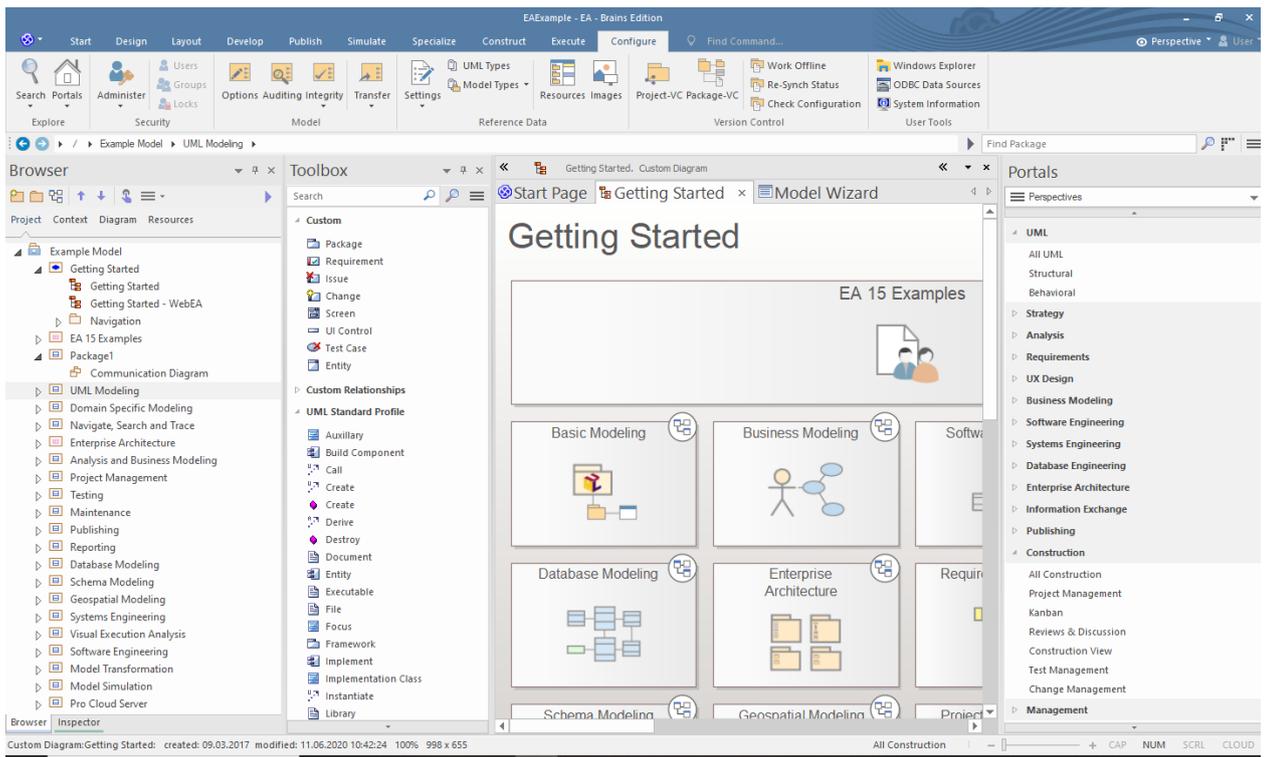
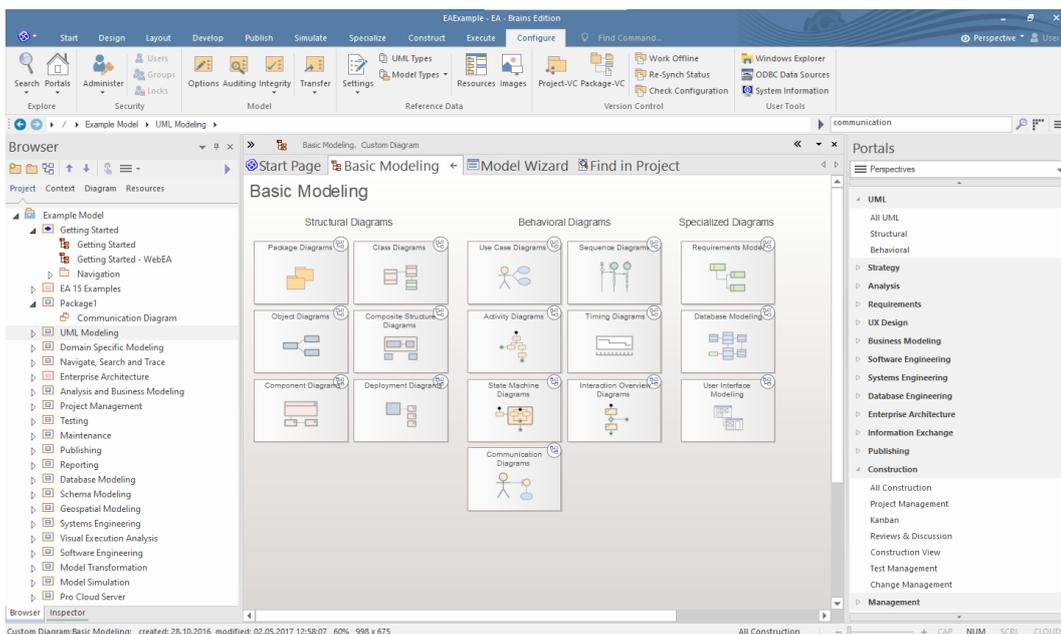
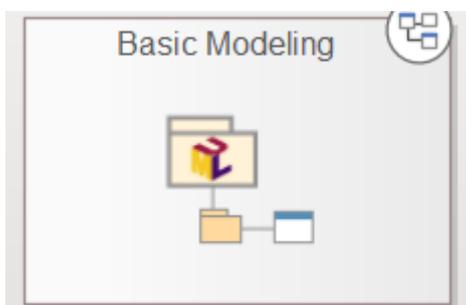


Диаграмма синхронизции (timing diagram)

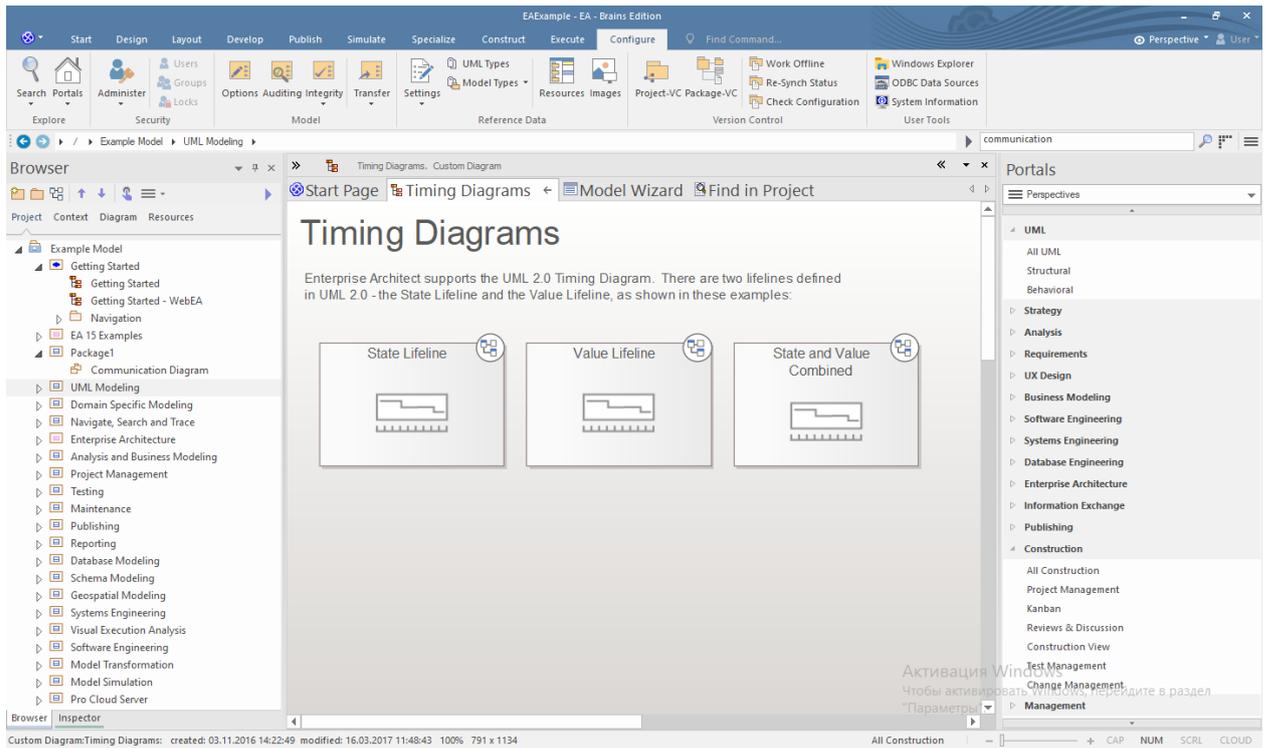
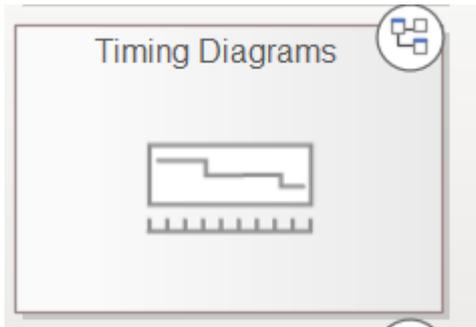
При входе в Enterprise arc. В меню выбрать вкладку Getting Started



На фото выше выбираем вкладку Basic Modeling (базовое моделирование)



Далее, во вкладке базовых моделей выбираем нужную нам timing diagrams



Пример state lifeline:

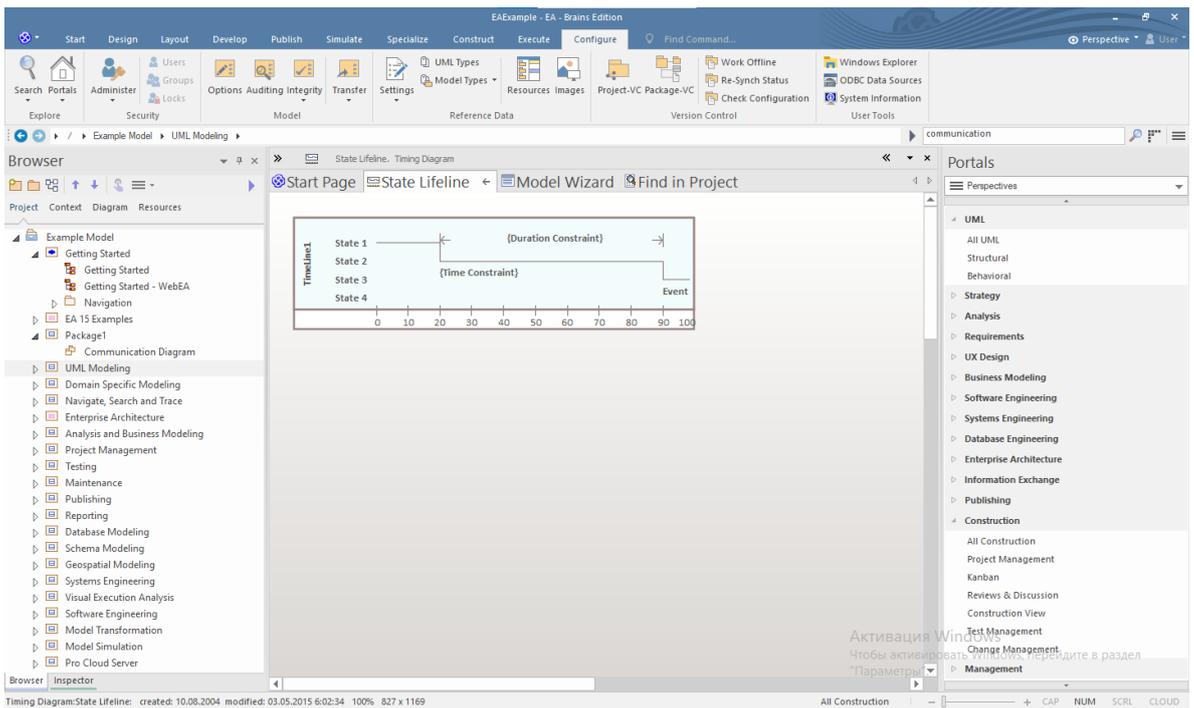
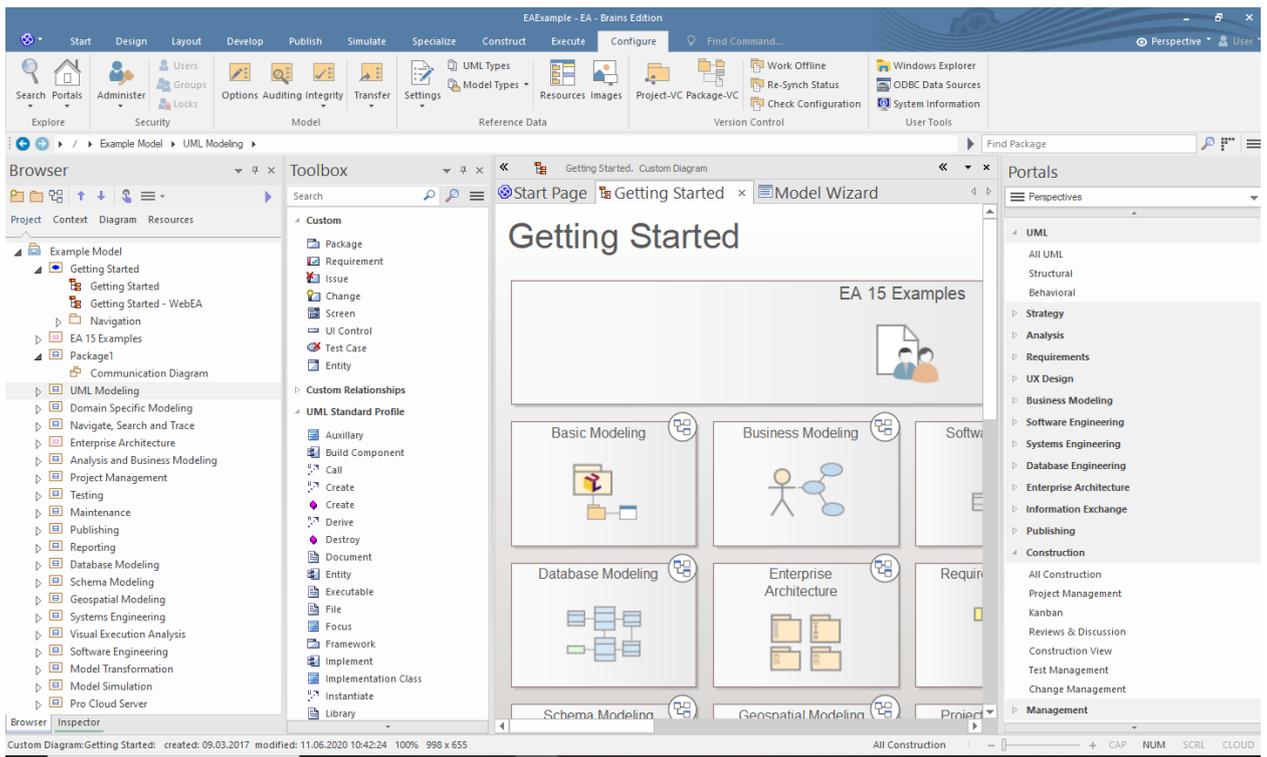
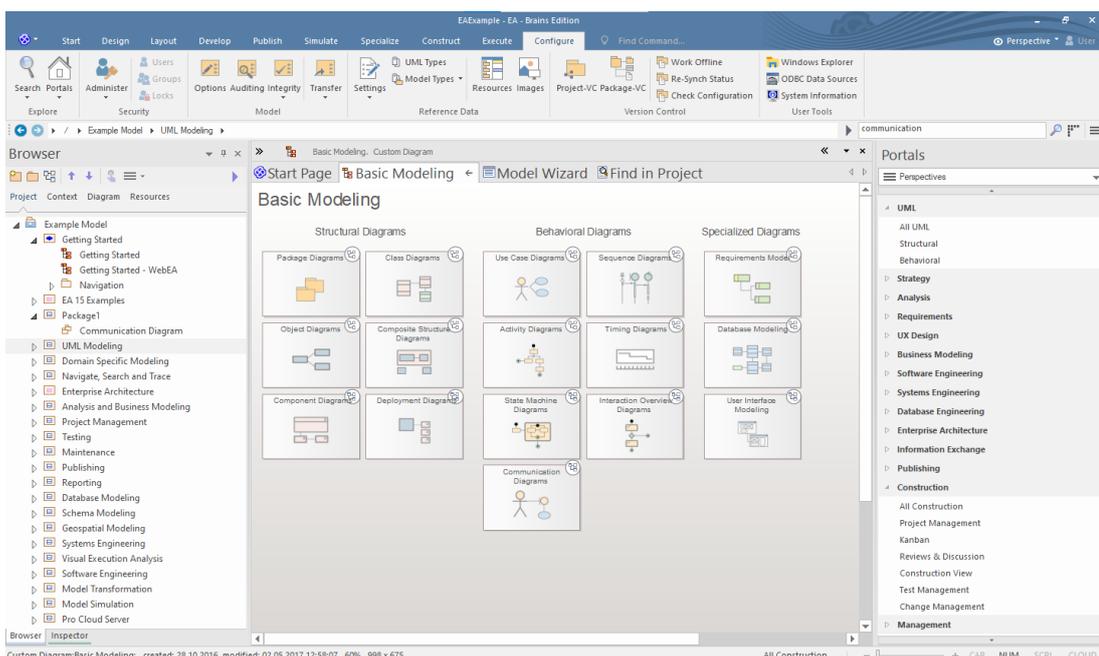
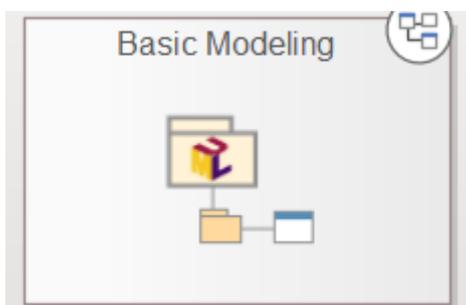


Диаграмма классов(Class diagram)

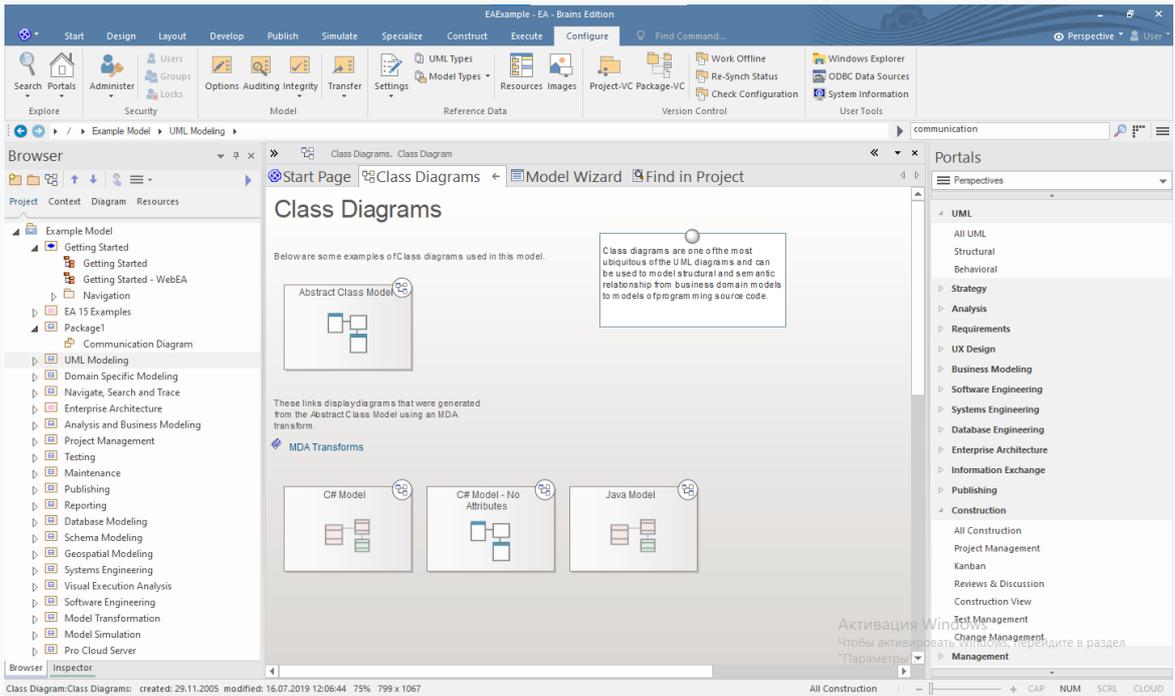
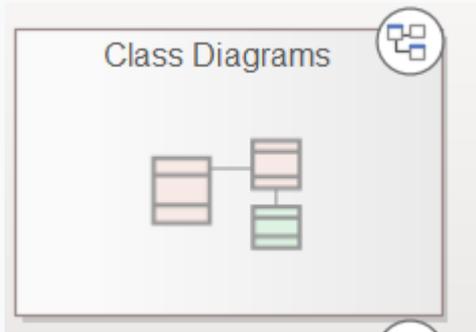
При входе в Enterprise arc. В меню выбрать вкладку Getting Started



На фото выше выбираем вкладку Basic Modeling (базовое моделирование)



Далее, во вкладке базовых моделей выбираем нужную нам Class Diagrams



Class Diagrams

Below are some examples of class diagrams used in this model.

Class diagrams are one of the most ubiquitous of the UML diagrams and can be used to model structural and semantic relationships from business domain models to models of programming source code.

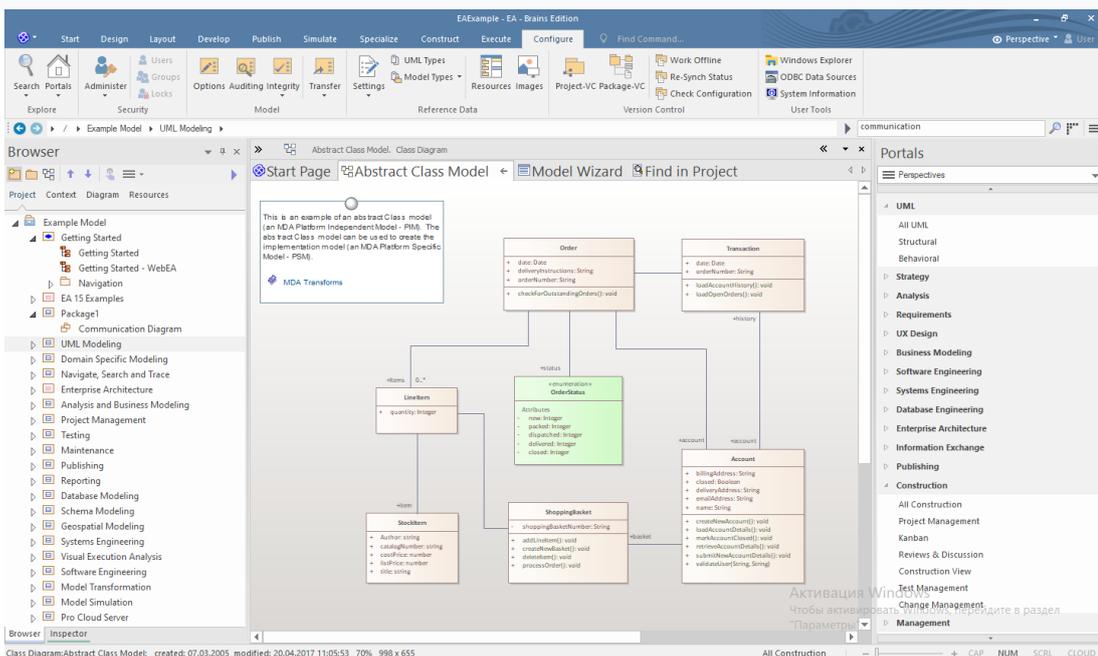
These links display diagrams that were generated from the Abstract Class Model using an MDA transform.

MDA Transforms

- Abstract Class Model
- C# Model
- C# Model - No Attributes
- Java Model

Class Diagram:Class Diagrams: created: 29.11.2005 modified: 16.07.2019 12:06:44 75% 799 x 1067

Пример абстрактного класса-модели



This is an example of an abstract Class model (an MDA Platform Independent Model - PIM). The abstract Class model can be used to create the implementation model (an MDA Platform Specific Model - PSM).

MDA Transforms

- Order
- Transaction
- LineItem
- ShoppingBasket
- Account

Class Diagram:Abstract Class Model: created: 07.03.2005 modified: 20.04.2017 11:05:53 70% 998 x 655

Таблица базы данных

UML-профиль Enterprise Architect для моделирования данных представляет: · Таблицы базы данных как объекты класса UML со стереотипом <<table>>

· Столбцы таблицы как UML-атрибуты таблицы со стереотипом <<column>>

Первичные ключи как UML-операции/методы таблицы, имеющие стереотип <<PK>>

· Внешние ключи как UML-операции/методы таблицы, имеющие стереотип <<FK>>

· Индексы - это UML-операции/методы таблицы, имеющие стереотип <<index>>

· Уникальные ограничения - это UML-операции/методы таблицы, имеющие стереотип

· Проверьте ограничения как UML-операции/методы таблицы, используя стереотип

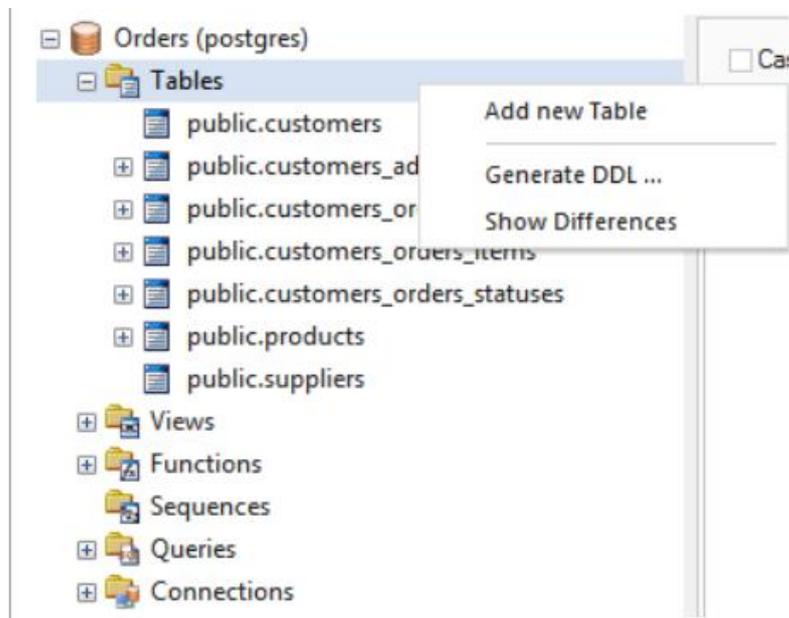
Табличные триггеры - это UML-операции/методы таблицы со стереотипом <<trigger>>

Добавить таблицу базы данных

1. Откройте конструктор баз данных (Design > Tools > Database Builder).

2 Загрузите или создайте модель данных.

3 Щелкните правой кнопкой мыши на пакете таблиц и выберите "Добавить новую таблицу".



4 Вместо имени по умолчанию введите соответствующее имя таблицы и нажмите клавишу Enter.

5 Дважды щелкните по элементу таблицы, чтобы задать свойства таблицы.

Создание столбцов таблицы базы данных

Ribbon	Design > Tools > Database Builder > Click on Table > Columns > Right-Click > Add new Column
--------	---

Menu	Tools Database Builder Click on Table > Columns Right-Click Add new Column Select a table F9 Tab Key
	На диаграмме щелкните правой кнопкой мыши на требуемой таблице Features & Properties Attributes Columns Right-Click Add new Colum
	Выбрать таблицу F9 Tab Key

Удаление столбцов таблицы базы данных

Ribbon	Design > Tools > Database Builder > Click on Table > Columns > нажать правой кнопкой мыши на название столбца > Delete column
Menu	Tools Database Builder Click on Table > Columns нажать правой кнопкой мыши на название столбца Delete column <name>
	На диаграмме щелкните правой кнопкой мыши на требуемой таблице Features & Properties Attributes > Columns Right-click on column name Delete column
	F9 Use 'Up Arrow' or 'Down Arrow' для выбора столбца Ctrl +

Изменить порядок столбцов

1. На вкладке «Columns» щелкните по названию нужного столбца в списке.
2. Щелкните правой кнопкой мыши и выберите: параметр «Move column <name> up»(или нажмите Ctrl+ стрелка вверх), чтобы переместить столбец на одну позицию вверх.
3. Щелкните правой кнопкой мыши и выберите: параметр «Move column <name> down» (или нажмите Ctrl+стрелка вниз), чтобы переместить столбец на одну позицию вниз.

Управление параметрами СУБД

	Отображает название пакета в браузере проекта, над которым вы в данный момент работаете. При необходимости нажмите на кнопку  и выберите другой пакет.
Include Objects in Child Packages	установите этот флажок, чтобы включить все объекты базы данных во все подпакеты. При выборе или отмене выбора этого элемента управления список объектов будет немедленно обновлен.
	Этот элемент управления списком будет отображать имена всех объектов в текущем пакете (или Иерархия пакетов) вместе с выделенной для них СУБД и

	<p>владельцем. По умолчанию флажок установлен для каждого объекта при каждой загрузке или обновлении списка.</p>
1	<p>Нажмите на эту кнопку, чтобы установить все снятые флажки в 'List of</p>
	<p>Нажмите на эту кнопку, чтобы снять все установленные флажки в 'List of Object</p>
C	<p>Установите этот флажок, если вы хотите изменить назначенную СУБД для объектов в Пакете. Чтобы продолжить, укажите значения в полях "Current DBMS" и "New DBMS" . В раскрывающемся списке изменяет несколько различных значений СУБД на новое значение. Примечание: При выполнении этой функции типы данных всех столбцов таблицы автоматически преобразуются в наиболее подходящие для выбранной СУБД; поэтому после запуска процедуры вам следует выполнить проверку типов данных вручную.</p>
	<p>Установите этот флажок, если вы хотите изменить владельца выбранных объектов в " List of Objects ". Чтобы продолжить, укажите текущего владельца в поле " Current Owner ". Если оставить поле "Новый владелец" пустым, свойство " New Owner " будет удалено для всех выбранных объектов.</p>
Create Indexes on Foreign Keys	<p>Установите этот флажок, чтобы создать индекс для всех внешних ключей в пакете, если они еще не используются.</p>
	<p>Нажмите на эту кнопку, чтобы начать процесс обновления. Кнопка отключает по крайней мере один объект в списке и один из вариантов обновления</p>

Логическая модель данных



Обратите внимание, что элементы данных "Клиенты" и "Адреса клиентов" содержат атрибуты UML; имена и общие типы данных должны оставаться независимыми от платформы. Типы данных, относящиеся к конкретной платформе, и другие метаданные, относящиеся к конкретной СУБД реализация определяется физической моделью данных

Физическая модель данных

