**Лекция 7:**

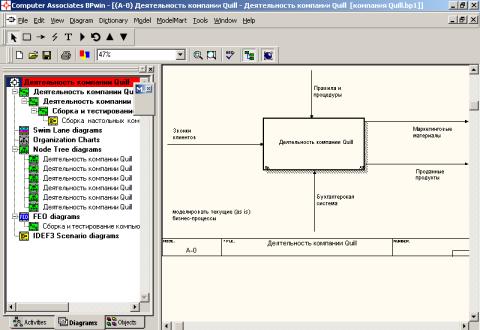
**Моделирование бизнес-процессов средствами BPwin**

### Инструментальная среда BPwin

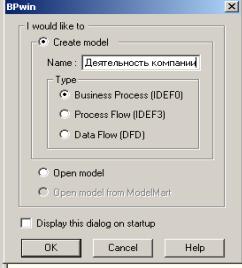
BPwin имеет достаточно простой и интуитивно понятный *интерфейс* пользователя. При запуске BPwin по умолчанию появляется основная *панель инструментов*, палитра инструментов (вид которой зависит от выбранной нотации) и, в левой части, навигатор модели — *Model* Explorer ([рис. 7.1](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=1#image.7.1)).

При создании новой модели возникает диалог, в котором следует указать, будет ли создана модель заново или она будет открыта из файла либо из репозитория ModelMart, затем внести имя модели и выбрать методологию, в которой будет построена модель ([рис. 7.2](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=1#image.7.2)).

Как было указано выше, BPwin поддерживает три методологии — *IDEF0*, IDEF3 и *DFD*, каждая из которых решает свои специфические задачи. В BPwin возможно построение смешанных моделей, т. е. модель может содержать одновременно диаграммы как *IDEF0*, так и IDEF3 и *DFD*. Состав палитры инструментов изменяется автоматически, когда происходит переключение с одной нотации на другую.



**Рис. 7.1.**Интегрированная среда разработки модели BPwin



**Рис. 7.2.**Диалог создания модели

Модель в BPwin рассматривается как совокупность *работ*, каждая из которых оперирует с некоторым набором данных. *Работа* изображается в виде прямоугольников, данные — в виде стрелок. Если щелкнуть по любому объекту модели левой кнопкой мыши, появляется контекстное *меню*, каждый *пункт* которого соответствует редактору какого-либо свойства объекта.

### Построение модели IDEF0

На начальных *этапах создания ИС* необходимо понять, как работает организация, которую собираются автоматизировать. Руководитель хорошо знает *работу* в целом, но не в состоянии вникнуть в детали *работы* каждого рядового сотрудника. Рядовой сотрудник хорошо знает, что творится на его рабочем месте, но может не знать, как работают коллеги. Поэтому для описания *работы* предприятия необходимо построить модель, которая будет адекватна *предметной области* и содержать в себе знания всех участников бизнес-процессов организации.

Наиболее удобным *языком моделирования* бизнес-процессов является *IDEF0*, где система представляется как совокупность взаимодействующих *работ* или функций. Такая чисто функциональная ориентация является принципиальной — функции системы анализируются независимо от объектов, которыми они оперируют. Это позволяет более четко смоделировать логику и взаимодействие процессов организации.

Процесс моделирования системы в *IDEF0* начинается с создания *контекстной диаграммы* — диаграммы наиболее абстрактного уровня описания системы в целом, содержащей *определение* субъекта моделирования, цели и точки зрения на модель.

Под субъектом понимается сама система, при этом необходимо точно установить, что входит в систему, а что лежит за ее пределами, другими словами, определить, что будет в дальнейшем рассматриваться как компоненты системы, а что как внешнее воздействие. На *определение* *субъекта системы* будут существенно влиять позиция, с которой рассматривается система, и *цель моделирования* — вопросы, на которые построенная модель должна дать ответ. Другими словами, в начале необходимо определить область моделирования. Описание области как системы в целом, так и ее компонентов является основой построения модели. Хотя предполагается, что в ходе моделирования область может корректироваться, она должна быть в основном сформулирована изначально, поскольку именно область определяет направление моделирования. При формулировании области необходимо учитывать два компонента — широту и глубину. Широта подразумевает *определение* границ модели — что будет рассматриваться внутри системы, а что снаружи. Глубина определяет, на каком уровне детализации модель является завершенной. При определении глубины системы необходимо помнить об ограничениях времени — трудоемкость построения модели растет в геометрической прогрессии с увеличением глубины декомпозиции. После определения границ модели предполагается, что новые объекты не должны вноситься в моделируемую систему.

#### Цель моделирования

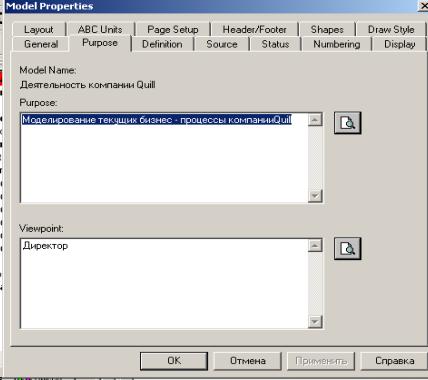
*Цель моделирования* определяется из ответов на следующие вопросы:

* Почему этот процесс должен быть смоделирован?
* Что должна показывать модель?
* Что может получить клиент?

Точка зрения (Viewpoint).

Под точкой зрения понимается перспектива, с которой наблюдалась система при построении модели. Хотя при построении модели учитываются мнения различных людей, все они должны придерживаться единой точки зрения на модель. Точка зрения должна соответствовать цели и границам моделирования. Как правило, выбирается точка зрения человека, ответственного за моделируемую *работу* в целом.

*IDEF0*-модель предполагает наличие четко сформулированной цели, единственного субъекта моделирования и одной точки зрения. Для внесения области, цели и точки зрения в модели *IDEF0* в BPwin следует выбрать пункт меню Model/Model Properties, вызывающий диалог Model Properties ([рис. 7.3](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=1#image.7.3)). В закладке Purpose следует внести цель и точку зрения, а в закладку Definition — определение модели и описание области.



**Рис. 7.3.**Диалог задания свойств модели

В закладке Status того же диалога можно описать статус модели (черновой вариант, рабочий, окончательный и т. д.), время создания и последнего редактирования (отслеживается в дальнейшем автоматически по системной дате). В закладке Source описываются источники информации для построения модели (например, "Опрос экспертов предметной области и анализ документации"). Закладка General служит для внесения имени проекта и модели, имени и инициалов автора и временных рамок модели — AS-IS и ТО-ВЕ.

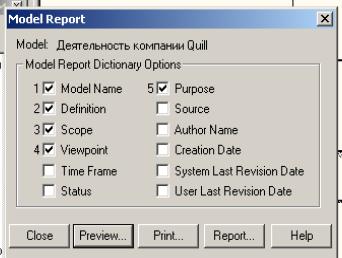
Модели AS-IS и ТО-ВЕ. Обычно сначала строится модель существующей организации *работы* — AS-IS (как есть). Анализ *функциональной модели* позволяет понять, где находятся наиболее слабые места, в чем будут состоять преимущества новых бизнес-процессов и насколько глубоким изменениям подвергнется существующая *структура организации* бизнеса. Детализация бизнес-процессов позволяет выявить недостатки организации даже там, где функциональность на первый взгляд кажется очевидной. Найденные в модели AS-IS недостатки можно исправить при создании модели ТО-ВЕ (как будет) — модели новой организации бизнес-процессов.

Технология *проектирования ИС* подразумевает сначала создание модели AS-IS, ее анализ и улучшение бизнес-процессов, то есть создание модели ТО-ВЕ, и только на основе модели ТО-ВЕ строится модель данных, прототип и затем окончательный вариант ИС.

Иногда текущая AS-IS и будущая ТО-ВЕ модели различаются очень сильно, так что переход от начального к конечному состоянию становится неочевидным. В этом случае необходима третья модель, описывающая процесс перехода от начального к конечному состоянию системы, поскольку такой переход — это тоже бизнес-процесс.

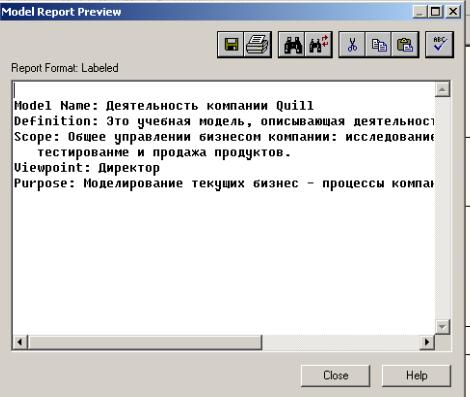
Результат описания модели можно получить в отчете Model Report. Диалог настройки отчета по модели вызывается из пункта меню Tools/Reports/Model Report.

В диалоге настройки следует выбрать необходимые поля, при этом автоматически отображается очередность вывода информации в отчет ([рис. 7.4](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=1#image.7.4)).



**Рис. 7.4.**Диалоговое окно для формирования отчета по модели

На [рис. 7.5](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=1#image.7.5) представлен отчет, сформированный по вышеуказанным полям.



**Рис. 7.5.**Предварительный просмотр отчета

**[Дал >>](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2)**

Основу методологии *IDEF0* составляет графический язык описания бизнес-процессов. Модель в нотации *IDEF0* представляет собой совокупность иерархически упорядоченных и взаимосвязанных диаграмм. Каждая диаграмма является единицей описания системы и располагается на отдельном листе.

Модель может содержать четыре типа диаграмм:

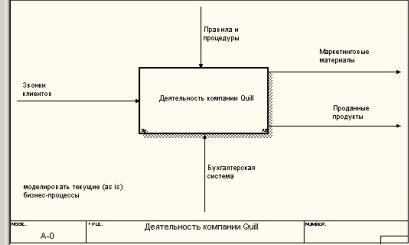
* *контекстную диаграмму* (в каждой модели может быть только одна *контекстная диаграмма* );
* *диаграммы декомпозиции*;
* *диаграммы дерева узлов* ;
* *диаграммы только для экспозиции (FEO)*.

*Контекстная диаграмма* является вершиной древовидной структуры диаграмм и представляет собой самое общее описание системы и ее взаимодействия с внешней средой. После описания системы в целом проводится разбиение ее на крупные фрагменты. Этот процесс называется функциональной декомпозицией, а диаграммы, которые описывают каждый фрагмент и взаимодействие фрагментов, называются диаграммами декомпозиции. После декомпозиции *контекстной диаграммы* проводится декомпозиция каждого большого фрагмента системы на более мелкие и так далее, до достижения нужного уровня подробности описания. После каждого сеанса декомпозиции проводятся сеансы экспертизы — эксперты предметной области указывают на соответствие реальных бизнес-процессов созданным диаграммам. Найденные несоответствия исправляются, и только после прохождения экспертизы без замечаний можно приступать к следующему сеансу декомпозиции. Так достигается соответствие модели реальным бизнес-процессам на любом и каждом уровне модели. Синтаксис описания системы в целом и каждого ее фрагмента одинаков во всей модели.

*Диаграмма дерева узлов* показывает иерархическую зависимость *работ*, но не взаимосвязи между *работами*. *Диаграмм деревьев узлов* может быть в модели сколь угодно много, поскольку дерево может быть построено на произвольную глубину и не обязательно с корня.

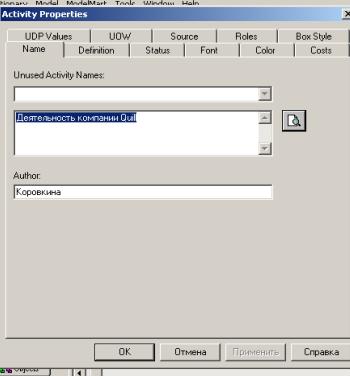
*диаграммы для экспозиции (FEO)* строятся для иллюстрации отдельных фрагментов модели, для иллюстрации альтернативной точки зрения, либо для специальных целей.

*Работы* **(Activity)** обозначают поименованные процессы, функции или задачи, которые происходят в течение определенного времени и имеют распознаваемые результаты. *Работы* изображаются в виде прямоугольников. Все *работы* должны быть названы и определены. Имя *работы* должно быть выражено отглагольным существительным, обозначающим действие (например, "Деятельность компании", "Прием заказа" и т.д.). *Работа* "Деятельность компании" может иметь, например, следующее определение: "Это учебная модель, описывающая деятельность компании". При создании новой модели (меню File/New) автоматически создается *контекстная диаграмма* с единственной *работой*, изображающей систему в целом ([рис. 7.6](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.6)).



**Рис. 7.6.**Пример контекстной диаграммы

Для внесения имени *работы* следует щелкнуть по *работе* правой кнопкой мыши, выбрать в меню Name Editor и в появившемся диалоге внести имя *работы*. Для описания других свойств *работы* служит диалог Activity Properties ([рис. 7.7](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.7)).



**Рис. 7.7.**Редактор задания свойств работы

*Диаграммы декомпозиции* содержат родственные *работы*, т. е. дочерние *работы*, имеющие общую родительскую *работу*. Для создания *диаграммы декомпозиции* следует щелкнуть по кнопке

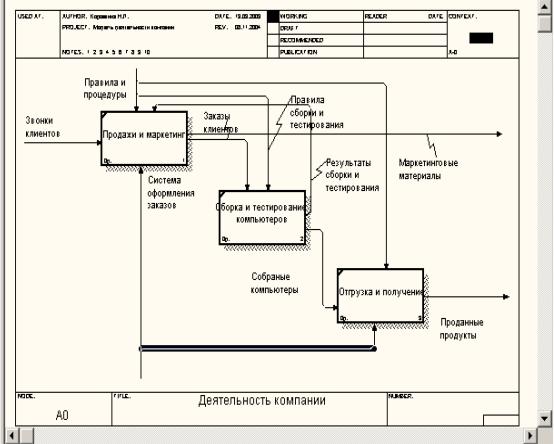
https://intuit.ru/EDI/25_07_20_1/1595629193-15985/tutorial/134/objects/7/files/kno.jpg

на панели инструментов.

Возникает диалог Activity Box Count ([рис. 7.8](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.8)), в котором следует указать нотацию новой диаграммы и количество *работ* на ней. Остановимся пока на нотации *IDEF0* и щелкнем на ОК. Появляется диаграмма декомпозиции ([рис. 7.9](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.9)). Допустимый интервал числа *работ* — 2-8. Декомпозировать *работу* на одну *работу* не имеет смысла: диаграммы с количеством *работ* более восьми получаются перенасыщенными и плохо читаются. Для обеспечения наглядности и лучшего понимания моделируемых процессов рекомендуется использовать от трех до шести блоков на одной диаграмме.



**Рис. 7.8.**Диалог Activity Box Count



**Рис. 7.9.**Пример диаграммы декомпозиции

Если оказывается, что количество *работ* недостаточно, то *работу* можно добавить в диаграмму, щелкнув сначала по кнопке на палитре инструментов, а затем по свободному месту на диаграмме.

*Работы* на диаграммах декомпозиции обычно располагаются по диагонали от левого верхнего угла к правому нижнему.

Такой порядок называется порядком доминирования. Согласно этому принципу расположения в левом верхнем углу помещается самая важная *работа* или *работа*, выполняемая по времени первой. Далее вправо вниз располагаются менее важные или выполняемые позже *работы*. Такое размещение облегчает чтение диаграмм, кроме того, на нем основывается понятие взаимосвязей *работ* (см. ниже).

Каждая из *работ* на *диаграмме декомпозиции* может быть в свою очередь декомпозирована. На *диаграмме декомпозиции* *работы* нумеруются автоматически слева направо. Номер *работы* показывается в правом нижнем углу. В левом верхнем углу изображается небольшая диагональная черта, которая показывает, что данная *работа* не была декомпозирована. Так, на [рис. 7.9](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.9) все *работы* еще не были декомпозированы. *Стрелки* (Arrow) описывают взаимодействие *работ* и представляют собой некую информацию, выраженную существительными.(Например, "Звонки клиентов", "Правила и процедуры", "Бухгалтерская система".)

В *IDEF0* различают пять типов стрелок:

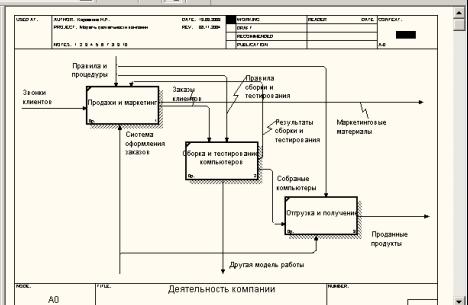
**Вход** ( **Input** ) — материал или информация, которые используются или преобразуются *работой* для получения результата (выхода). Допускается, что *работа* может не иметь ни одной *стрелки* входа. Каждый тип стрелок подходит к определенной стороне прямоугольника, изображающего *работу*, или выходит из нее. *Стрелка* входа рисуется как входящая в левую грань *работы*. При описании технологических процессов (для этого и был придуман *IDEF0*) не возникает проблем определения входов. Действительно, "Звонки клиентов" на [рис. 7.6](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.6) — это нечто, что перерабатывается в процессе "Деятельность компании" для получения результата. При моделировании ИС, когда *стрелками* являются не физические объекты, а данные, не все так очевидно. Например, при "Приеме пациента" карта пациента может быть и на входе и на выходе, между тем качество этих данных меняется. Другими словами, в нашем примере для того, чтобы оправдать свое назначение, *стрелки* входа и выхода должны быть точно определены с тем, чтобы указать на то, что данные действительно были переработаны (например, на выходе — "Заполненная карта пациента"). Очень часто сложно определить, являются ли данные входом или управлением. В этом случае подсказкой может служить информация о том, перерабатываются/изменяются ли данные в *работе* или нет. Если изменяются, то, скорее всего, это вход, если нет — управление.

**Управление** ( **Control** ) — правила, стратегии, процедуры или стандарты, которыми руководствуется *работа*. Каждая *работа* должна иметь хотя бы одну *стрелку* управления. *Стрелка* управления рисуется как входящая в верхнюю грань *работы*. На [рис. 7.6](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.6) *стрелка* "Правила и процедуры" — управление для *работы* "Деятельность компании". Управление влияет на *работу*, но не преобразуется *работой*. Если цель *работы* — изменить процедуру или стратегию, то такая процедура или стратегия будет для *работы* входом. В случае возникновения неопределенности в статусе *стрелки* (управление или вход) рекомендуется рисовать *стрелку* управления.

**Выход** ( **Output** ) — материал или информация, которые производятся *работой*. Каждая *работа* должна иметь хотя бы одну *стрелку* выхода. *Работа* без результата не имеет смысла и не должна моделироваться. *Стрелка* выхода рисуется как исходящая из правой грани *работы*. На [рис. 7.6](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.6) *стрелки* "Маркетинговые материалы" и "Проданные продукты" являются выходом для *работы* "Деятельность компании".

**Механизм** ( **Mechanism** ) — ресурсы, которые выполняют *работу*, например персонал предприятия, станки, устройства и т. д. *Стрелка* механизма рисуется как входящая в нижнюю грань *работы*. На [рис. 7.6](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.6) *стрелка* "Бухгалтерская система" является механизмом для *работы* "Деятельность компании". По усмотрению аналитика *стрелки* механизма могут не изображаться в модели.

**Вызов** ( **Call** ) — специальная *стрелка*, указывающая на другую модель *работы*. *Стрелка* вызова рисуется как исходящая из нижней грани *работы*. На [рис. 7.10](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=3#image.7.10) *стрелка* "Другая модель *работы* " является вызовом для *работы* "Изготовление изделия". *Стрелка* вызова используется для указания того, что некоторая *работа* выполняется за пределами моделируемой системы. В BPwin *стрелки* вызова используются в механизме слияния и разделения моделей.



**Рис. 7.10.**Стрелка вызова, появляющаяся при расщеплении модели

**Граничные** *стрелки*. *Стрелки* на *контекстной диаграмме* служат для описания взаимодействия системы с окружающим миром. Они могут начинаться у границы диаграммы и заканчиваться у *работы*, или наоборот. Такие *стрелки* называются граничными.

Для внесения граничной *стрелки* входа следует:

* щелкнуть по кнопке с символом *стрелки*

https://intuit.ru/EDI/25_07_20_1/1595629193-15985/tutorial/134/objects/7/files/kno1.jpg

;

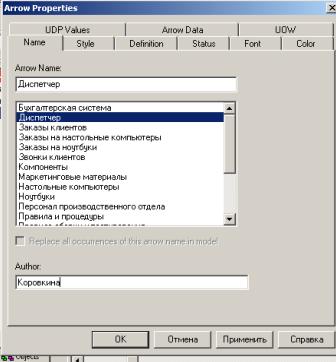
* в палитре инструментов перенести курсор к левой стороне экрана, пока не появится начальная штриховая полоска;
* щелкнуть один раз по полоске (откуда выходит *стрелка* ) и еще раз в левой части *работы* со стороны входа (где заканчивается *стрелка* );
* вернуться в палитру инструментов и выбрать опцию редактирования *стрелки*

https://intuit.ru/EDI/25_07_20_1/1595629193-15985/tutorial/134/objects/7/files/kno2.jpg

;

* щелкнуть правой кнопкой мыши на линии *стрелки*, во всплывающем меню выбрать Name и добавить имя *стрелки* в закладке Name диалога *IDEF0* Arrow Properties.

*Стрелки* управления, входа, механизма и выхода изображаются аналогично. Имена вновь внесенных стрелок ([рис. 7.11](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=3#image.7.11)) автоматически заносятся в словарь Arrow Dictionary.

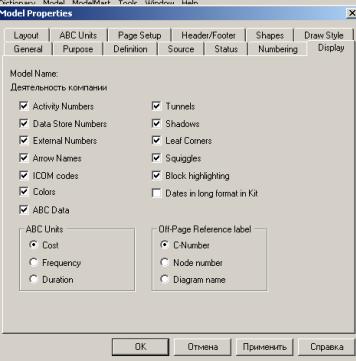


**Рис. 7.11.**Диалог IDEF0 Arrow Properties

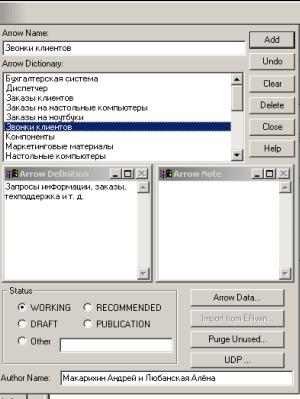
**ICOM-коды**. Диаграмма декомпозиции предназначена для детализации *работы*. В отличие от моделей, отображающих *структуру организации*, *работа* на диаграмме верхнего уровня в *IDEF0* — это не элемент управления нижестоящими *работами*. *Работы* нижнего уровня — это то же самое, что *работы* верхнего уровня, но в более детальном изложении. Как следствие этого границы *работы* верхнего уровня — это то же самое, что границы *диаграммы декомпозиции*. ICOM (аббревиатура от Input, Control, Output и *Mechanism*) — коды, предназначенные для идентификации граничных стрелок. Код ICOM содержит префикс, соответствующий типу *стрелки* (I, С, О или М), и порядковый номер.

BPwin вносит ICOM-коды автоматически. Для отображения ICOM-кодов следует включить опцию ICOM codes на закладке Display диалога Model Properties (меню Model/Model Properties) ([рис.7.12](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=3#image.7.12)).

Словарь **стрелок** редактируется при помощи специального редактора Arrow Dictionary Editor, в котором определяется *стрелка* и вносится относящийся к ней комментарий ([рис. 7.13](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=3#image.7.13)). Словарь стрелок решает очень важную задачу. Диаграммы создаются аналитиком для того, чтобы провести сеанс экспертизы, т. е. обсудить диаграмму со *специалистом предметной области*. В любой предметной области формируется профессиональный жаргон, причем очень часто жаргонные выражения имеют нечеткий смысл и воспринимаются разными специалистами по-разному. В то же время аналитик — автор диаграмм должен употреблять те выражения, которые наиболее понятны экспертам. Поскольку формальные определения часто сложны для восприятия, аналитик вынужден употреблять профессиональный жаргон, а чтобы не возникло неоднозначных трактовок, в словаре стрелок каждому понятию можно дать расширенное и, если это необходимо, формальное определение.



**Рис. 7.12.**Включение опции ICOM codes на закладке Display

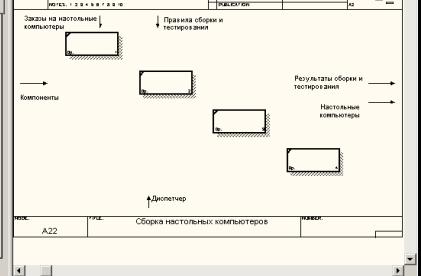


**Рис. 7.13.**Редактирование словаря стрелок

Содержимое словаря стрелок можно распечатать в виде отчета (меню Tools/ Report /Arrow Report...) и получить толковый словарь терминов предметной области, использующихся в модели.

**Несвязанные граничные** *стрелки* **(unconnected border arrow)**. При декомпозиции *работы* входящие в нее и исходящие из нее *стрелки* (кроме *стрелки* вызова) автоматически появляются на *диаграмме декомпозиции* (миграция стрелок), но при этом не касаются *работ*. Такие *стрелки* называются несвязанными и воспринимаются в BPwin как синтаксическая ошибка.

На [рис. 7.14](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=3#image.7.14) приведен фрагмент *диаграммы декомпозиции* с несвязанными *стрелками*, генерирующийся BPwin при декомпозиции *работы* **"Сборка настольных компьютеров"** (см. [рис. 7.9](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=2#image.7.9)). Для связывания стрелок входа, управления или механизма необходимо перейти в режим редактирования стрелок, щелкнуть по наконечнику *стрелки* и потом по соответствующему сегменту *работы*. Для связывания *стрелки* выхода необходимо перейти в режим редактирования стрелок, щелкнуть по сегменту выхода *работы* и затем по *стрелке*.



**Рис. 7.14.**Пример несвязанных стрелок

**Внутренние** *стрелки*. Для связи *работ* между собой используются внутренние *стрелки*, то есть *стрелки*, которые не касаются границы диаграммы, начинаются у одной и кончаются у другой *работы*.

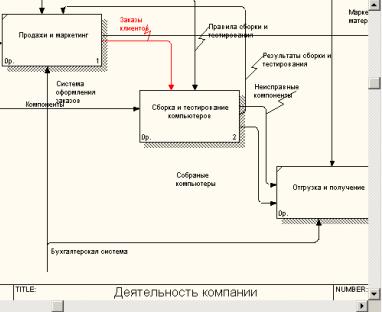
Для рисования внутренней *стрелки* необходимо в режиме рисования стрелок щелкнуть по сегменту (например, выхода) одной *работы* и затем по сегменту (например, входа) другой. В *IDEF0* различают пять *типов связей* *работ*.

**Связь по входу** ( **output-input** ), когда *стрелка* выхода вышестоящей *работы* (далее — просто выход) направляется на вход нижестоящей (например, на [рис. 7.15](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.15) *стрелка* **"Собранные компьютеры"** связывает *работы* **"Сборка и тестирование компьютеров"** и **"Отгрузка и получение"** ).



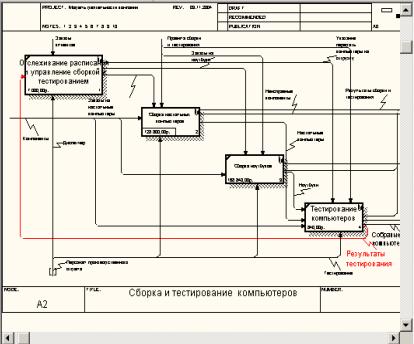
**Рис. 7.15.**Связь по входу

**Связь по управлению** ( **output-control** ), когда выход вышестоящей *работы* направляется на управление нижестоящей. Связь по управлению показывает доминирование вышестоящей *работы*. Данные или объекты выхода вышестоящей *работы* не меняются в нижестоящей. На [рис. 7.16](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.16) *стрелка* **"Заказы клиентов"** связывает *работы* **"Продажи и маркетинг"** и **"Сборка и тестирование компьютеров"**.



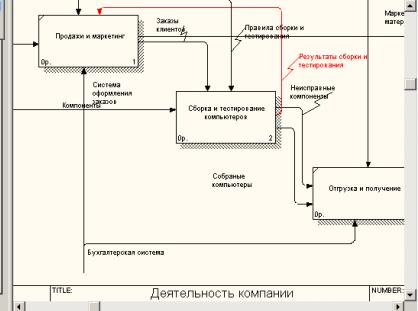
**Рис. 7.16.**Связь по управлению

**Обратная связь по входу** ( **output-input feedback** ), когда выход нижестоящей *работы* направляется на вход вышестоящей. Такая связь, как правило, используется для описания циклов. На [рис. 7.17](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.17) *стрелка* "Результаты тестирования" связывает *работы* **"Тестирование компьютеров"** и **"Отслеживание расписания и управление сборкой и тестированием"**.



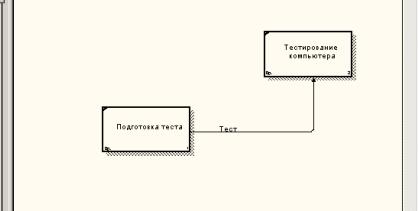
**Рис. 7.17.**Обратная связь по входу

**Обратная связь по управлению** ( **output-control feedback** ), когда выход нижестоящей *работы* направляется на управление вышестоящей ( *стрелка* "Результаты сборки и тестирования", [рис. 7.18](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.18)). Обратная связь по управлению часто свидетельствует об эффективности бизнес-процесса. На [рис. 7.18](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.18) объем продаж может быть повышен путем непосредственного регулирования процессов сборки и тестирования компьютеров (выхода) *работы* "Сборки и тестирование компьютеров".



**Рис. 7.18.**Обратная связь по управлению

**Связь выход-механизм** ( **output-mechanism** ), когда выход одной *работы* направляется на механизм другой. Эта взаимосвязь используется реже остальных и показывает, что одна *работа* подготавливает ресурсы, необходимые для проведения другой *работы* ([рис. 7.19](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.19)).

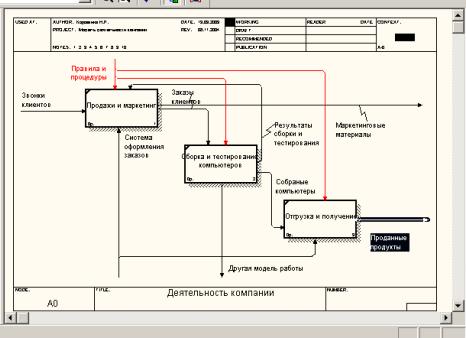


**Рис. 7.19.**Связь выход-механизм

**Явные** *стрелки*. Явная *стрелка* имеет источником одну-единственную *работу* и назначением тоже одну-единственную *работу*.

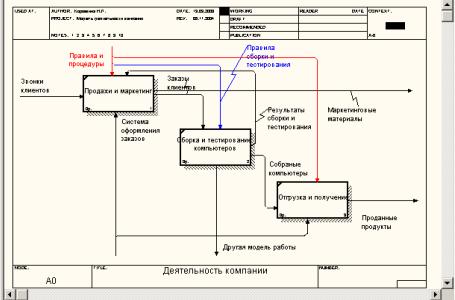
**Разветвляющиеся и сливающиеся** *стрелки*. Одни и те же данные или объекты, порожденные одной *работой*, могут использоваться сразу в нескольких других *работах*. С другой стороны, *стрелки*, порожденные в разных *работах*, могут представлять собой одинаковые или однородные данные или объекты, которые в дальнейшем используются или перерабатываются в одном месте. Для моделирования таких ситуаций в *IDEF0* используются разветвляющиеся и сливающиеся *стрелки*. Для разветвления *стрелки* нужно в режиме редактирования *стрелки* щелкнуть по фрагменту *стрелки* и по соответствующему сегменту *работы*. Для слияния двух стрелок выхода нужно в режиме редактирования *стрелки* сначала щелкнуть по сегменту выхода *работы*, а затем по соответствующему фрагменту *стрелки*.

Смысл разветвляющихся и сливающихся стрелок передается именованием каждой ветви стрелок. Существуют определенные правила именования таких стрелок. Рассмотрим их на примере разветвляющихся стрелок. Если *стрелка* именована до разветвления, а после разветвления ни одна из ветвей не именована, то подразумевается, что каждая ветвь моделирует те же данные или объекты, что и ветвь до разветвления ([рис. 7.20](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.20)).



**Рис. 7.20.**Пример именования разветвляющейся стрелки

Если *стрелка* именована до разветвления, а после разветвления какая-либо из ветвей тоже именована, то подразумевается, что эти ветви соответствуют именованию. Если при этом какая-либо ветвь после разветвления осталась неименованной, то подразумевается, что она моделирует те же данные или объекты, что и ветвь до разветвления ([рис. 7.21](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.21)).



**Рис. 7.21.**Пример именования разветвляющейся стрелки

Недопустима ситуация, когда *стрелка* до разветвления не именована, а после разветвления не именована какая-либо из ветвей. BPwin определяет такую *стрелку* как синтаксическую ошибку.

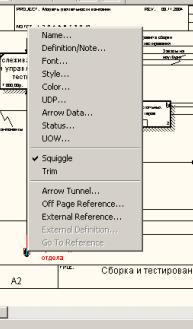
Правила именования сливающихся стрелок полностью аналогичны — ошибкой будет считаться *стрелка*, которая после слияния не именована, а до слияния не именована какая-либо из ее ветвей. Для именования отдельной ветви разветвляющихся и сливающихся стрелок следует выделить на диаграмме только одну ветвь, после чего вызвать редактор имени и присвоить имя *стрелке*. Это имя будет соответствовать только выделенной ветви.

**Туннелирование стрелок**. Вновь внесенные граничные *стрелки* на *диаграмме декомпозиции* нижнего уровня изображаются в квадратных скобках и автоматически не появляются на диаграмме верхнего уровня ([рис. 7.22](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.22)).



**Рис. 7.22.**Неразрешенная (unresolved) стрелка

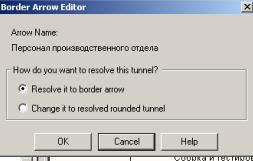
Для их "перетаскивания" наверх нужно щелкнуть правой кнопкой мыши по квадратным скобкам граничной *стрелки* и в контекстном меню выбрать команду Arrow *Tunnel* ([рис. 7.23](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.23)).



**Рис. 7.23.**Выбор команды из контекстного меню

Появляется диалог Border Arrow Editor ([рис. 7.24](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.24)).

Если щелкнуть по кнопке *Resolve* Border Arrow, *стрелка* мигрирует на диаграмму верхнего уровня, если по кнопке Change To *Tunnel* — *стрелка* будет туннелирована и не попадет на другую диаграмму. Туннельная *стрелка* изображается с круглыми скобками на конце ([рис. 7.25](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.25)).



**Рис. 7.24.**Диалог Border Arrow Editor



**Рис. 7.25.**Типы туннелирования стрелок

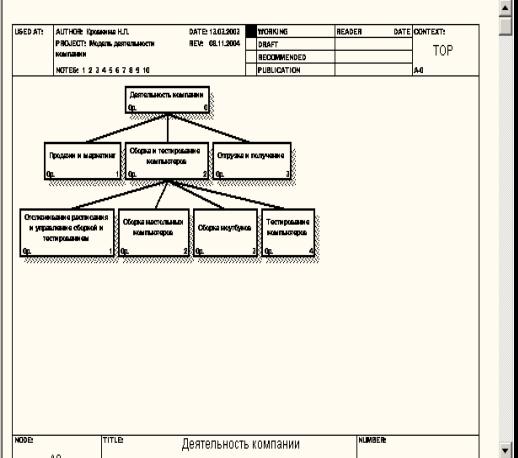
*Туннелирование* может быть применено для изображения малозначимых стрелок. Если на какой-либо диаграмме нижнего уровня необходимо изобразить малозначимые данные или объекты, которые не обрабатываются или не используются *работами* на текущем уровне, то их необходимо направить на вышестоящий уровень (на родительскую диаграмму). Если эти данные не используются на родительской диаграмме, их нужно направить еще выше, и т. д. В результате малозначимая *стрелка* будет изображена на всех уровнях и затруднит чтение всех диаграмм, на которых она присутствует. Выходом является *туннелирование* *стрелки* на самом нижнем уровне. Такое *туннелирование* называется "не-в-родительской-диаграмме".

Другим примером *туннелирования* может быть ситуация, когда *стрелка* механизма мигрирует с верхнего уровня на нижний, причем на нижнем уровне этот механизм используется одинаково во всех *работах* без исключения. (Предполагается, что не нужно детализировать *стрелку* механизма, т. е. *стрелка* механизма на дочерней *работе* именована до разветвления, а после разветвления ветви не имеют собственного имени). В этом случае *стрелка* механизма на нижнем уровне может быть удалена, после чего на родительской диаграмме она может быть туннелирована, а в комментарии к *стрелке* или в словаре можно указать, что механизм будет использоваться во всех *работах* дочерней *диаграммы декомпозиции*. Такое *туннелирование* называется "не-в-дочерней-работе" ([рис. 7.25](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=4#image.7.25)).

**Нумерация работ и диаграмм**. Все *работы* модели нумеруются. Номер состоит из префикса и числа. Может быть использован префикс любой длины, но обычно используют префикс А. Контекстная (корневая) *работа* дерева имеет номер А0. *Работы* i декомпозиции А0 имеют номера А1, А2, A3 и т. д. *Работы* декомпозиции нижнего уровня имеют номер родительской *работы* и очередной порядковый номер, например *работы* декомпозиции A3 будут иметь номера А31, А32, АЗЗ, А34 и т. д. *Работы* образуют иерархию, где каждая *работа* может иметь одну родительскую и несколько дочерних *работ*, образуя дерево. Такое дерево называют деревом узлов, а вышеописанную нумерацию — нумерацией по узлам. Диаграммы *IDEF0* имеют двойную нумерацию. Во-первых, диаграммы имеют номера по узлу. *Контекстная диаграмма* всегда имеет номер А-0, декомпозиция *контекстной диаграммы* — номер А0, остальные *диаграммы декомпозиции* — номера по соответствующему узлу (например, A1, A2, А21, А213 и т. д.). BPwin автоматически поддерживает нумерацию по узлам, т. е. при проведении декомпозиции создается новая диаграмма и ей автоматически присваивается соответствующий номер. В результате проведения экспертизы диаграммы могут уточняться и изменяться, следовательно, могут быть созданы различные версии одной и той же (с точки зрения ее расположения в дереве узлов) *диаграммы декомпозиции*. BPwin позволяет иметь в модели только одну *диаграмму декомпозиции* в данном узле. Прежние версии диаграммы можно хранить в виде бумажной копии либо как *FEO*-диаграмму. (К сожалению, при создании *FEO*-диаграмм отсутствует возможность отката, т. е. из диаграммы можно получить декомпозиции *FEO*, но не наоборот.) В любом случае следует отличать различные версии одной и той же диаграммы. Для этого существует специальный номер — C-number, который должен присваиваться автором модели вручную. C-number — это произвольная строка, но рекомендуется придерживаться стандарта, когда номер состоит из буквенного префикса и порядкового номера, причем в качестве префикса используются инициалы автора диаграммы, а порядковый номер отслеживается автором вручную, например МСВ00021

#### Диаграммы дерева узлов и FEO

***Диаграмма деревьев узлов*** показывает иерархию *работ* в модели и позволяет рассмотреть всю модель целиком, но не показывает взаимосвязи между *работами* ([рис. 7.26](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#image.7.26)).Процесс создания модели *работ* является итерационным, следовательно, *работы* могут менять свое расположение в дереве узлов многократно. Чтобы не запутаться и проверить способ декомпозиции, следует после каждого изменения создавать *диаграмму дерева узлов*. Впрочем, BPwin имеет мощный инструмент навигации по модели — Model Explorer, который позволяет представить иерархию *работ* и диаграмм в удобном и компактном виде.

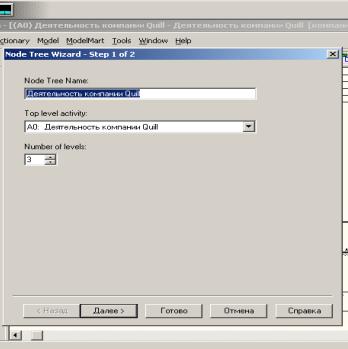


**Рис. 7.26.**Диаграмма дерева узлов

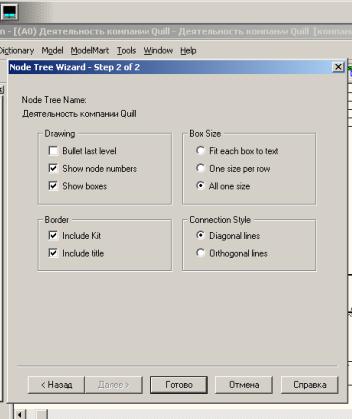
Для создания *диаграммы дерева узлов* следует выбрать в меню пункт Diagram/Add Node Tree ([рис. 7.27](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#image.7.27)). Возникает диалог формирования *диаграммы дерева узлов* Node Tree Definition ([рис. 7.28](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#image.7.28), [7. 29](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#image.7.29)).



**Рис. 7.27.**Выбор команды для формирования диаграммы дерева узлов



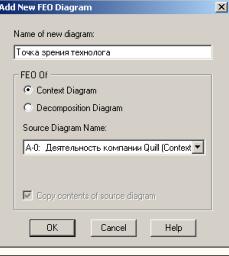
**Рис. 7.28.**Диалог настройки диаграммы дерева узлов (шаг 1)



**Рис. 7.29.**Диалог настройки диаграммы дерева узлов (шаг 2)

В диалоге Node Tree Definition следует указать глубину дерева — Number of Levels (по умолчанию — 3) и *корень дерева* (по умолчанию — родительская *работа* текущей диаграммы). По умолчанию нижний уровень декомпозиции показывается в виде списка, остальные *работы* — в виде прямоугольников. Для отображения всего дерева в виде прямоугольников следует выключить опцию *Bullet* Last Level. При создании дерева узлов следует указать имя диаграммы, поскольку, если в нескольких диаграммах в качестве корня на дереве узлов использовать одну и ту же *работу*, все эти диаграммы получат одинаковый номер (номер узла + постфикс N, например *AON*) и в списке открытых диаграмм (пункт меню Window) их можно будет различить только по имени.

*Диаграммы "только для экспозиции" (FEO)* часто используются в модели для иллюстрации других точек зрения, для отображения отдельных деталей, которые не поддерживаются явно синтаксисом *IDEF0*. Диаграммы *FEO* позволяют нарушить любое синтаксическое правило, поскольку по сути являются просто картинками — копиями стандартных диаграмм и не включаются в анализ синтаксиса. Для создания диаграммы *FEO* следует выбрать пункт меню Diagram/Add *FEO* Diagram. В возникающем диалоге Add New *FEO* Diagram следует указать имя диаграммы *FEO* и тип родительской диаграммы ([рис. 7.30](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#image.7.30)).

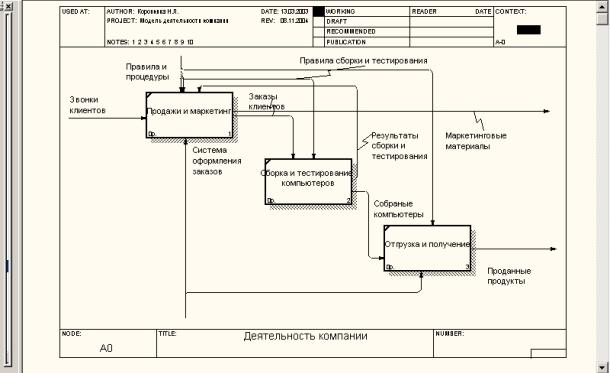


**Рис. 7.30.**Диалог создания FEO-диаграммы

Новая диаграмма получает номер, который генерируется автоматически (номер родительской диаграммы по узлу + постфикс F, например A1F ).

##### Каркас диаграммы

На [рис. 7.31](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#image.7.31) показан типичный пример *диаграммы декомпозиции* с граничными рамками, которые называются каркасом диаграммы.

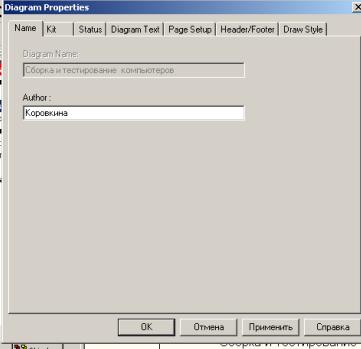


**Рис. 7.31.**Пример диаграммы декомпозиции с каркасом

Каркас содержит заголовок (верхняя часть рамки) и подвал (нижняя часть). Заголовок каркаса используется для отслеживания диаграммы в процессе моделирования. Нижняя часть используется для идентификации и позиционирования в иерархии диаграммы.

Смысл элементов каркаса приведен в [табл. 7.1](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#table.7.1) и [7.2](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=6#table.7.2).

Значения полей каркаса задаются в диалоге Diagram Properties (меню Diagram /Diagram Properties) — [рис. 7.32](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=5#image.7.32).



**Рис. 7.32.**Диалог Diagram Properties

|  |  |
| --- | --- |
| Таблица 7.1. Поля заголовка каркаса (слева направо) | |
| **Поле** | **Смысл** |
| Used At | Используется для указания на родительскую *работу* в случае, если на текущую диаграмму ссылались посредством *стрелки* вызова |
| Autor, Date, *Rev*, Project | Имя создателя диаграммы, дата создания и имя проекта, в рамках которого была создана диаграмма. *REV*-дата последнего редактирования диаграммы |
| Notes 123456789 10 | Используется при проведении сеанса экспертизы. Эксперт должен (на бумажной копии диаграммы) указать число замечаний, вычеркивая цифру из списка каждый раз при внесении нового замечания |
| Status | Статус отображает стадию создания диаграммы, отображая все этапы публикации |
| Working | Новая диаграмма, кардинально обновленная диаграмма или новый автор диаграммы |
| Draft | Диаграмма прошла первичную экспертизу и готова к дальнейшему обсуждению |
| Recommended | Диаграмма и все ее сопровождающие документы прошли экспертизу. Новых изменений не ожидается |
| Publication | Диаграмма готова к окончательной печати и публикации |
| Reader | Имя читателя (эксперта) |
| Date | Дата прочтения (экспертизы) |
| Context | Схема расположения *работ* в диаграмме верхнего уровня. *Работа*, являющаяся родительской, показана темным прямоугольником, остальные – светлым. На *контекстной диаграмме* (А-0) показана надпись ТОР. В левом нижнем углу показывается номер по узлу родительской диаграммы:  https://intuit.ru/EDI/25_07_20_1/1595629193-15985/tutorial/134/objects/7/files/table1.jpg |

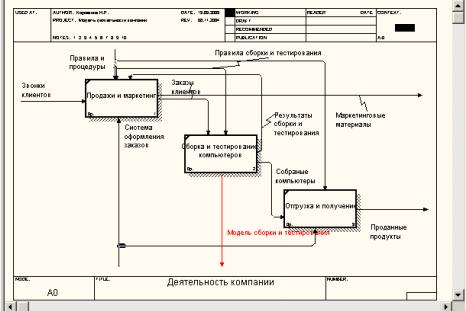
### Слияние и расщепление моделей

Возможность слияния и расщепления моделей обеспечивает коллективную *работу* над проектом. Так, *руководитель проекта* может создать декомпозицию верхнего уровня и дать задание аналитикам продолжить декомпозицию каждой ветви дерева в виде отдельных моделей. После окончания *работы* над отдельными ветвями все подмодели могут быть слиты в единую модель. С другой стороны, отдельная *ветвь* модели может быть отщеплена для использования в качестве независимой модели, для доработки или архивирования.

|  |  |
| --- | --- |
| Таблица 7.2. Поля подвала каркаса (слева направо) | |
| **Поле** | **Смысл** |
| Node | Номер узла диаграммы (номер родительской *работы* ) |
| Title | Имя диаграммы. По умолчанию — имя родительской *работы* |
| Number | C-Number, уникальный номер версии диаграммы |
| Page | Номер страницы, может использоваться как номер страницы при формировании папки |

BPwin использует для слияния и разветвления моделей *стрелки* вызова. Для слияния необходимо выполнить следующие условия:

* Обе сливаемые модели должны быть открыты в BPwin.
* Имя модели-источника, которое присоединяют к модели-цели, должно совпадать с именем *стрелки* вызова *работы* в модели-цели.
* *Стрелка* вызова должна исходить из недекомпозируемой *работы* ( *работа* должна иметь диагональную черту в левом верхнем углу) ([рис. 7.33](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=6#image.7.33)).

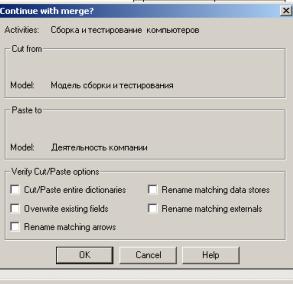


**Рис. 7.33.**Стрелка вызова работы "Сборка и тестирование компьютеров" модели-цели

* Имена контекстной *работы* подсоединяемой модели-источника и *работы* на модели-цели, к которой мы подсоединяем модель-источник, должны совпадать.
* Модель-источник должна иметь, по крайней мере, одну *диаграмму декомпозиции*.

Для слияния моделей нужно щелкнуть правой кнопкой мыши по *работе* со *стрелкой* вызова в модели-цели и во всплывающем *меню* выбрать *пункт* *Merge* *Model*.

Появляется диалог, в котором следует указать опции слияния модели ([рис. 7.34](https://intuit.ru/studies/courses/2195/55/lecture/1630?page=6#image.7.34)). При слиянии моделей объединяются и словари стрелок и *работ*. В случае одинаковых определений возможна перезапись определений или принятие определений из модели-источника. То же относится к именам стрелок, хранилищам данных и *внешним ссылкам*. (Хранилища данных и *внешние ссылки* — объекты *диаграмм потоков данных*, *DFD*, будут рассмотрены ниже.)



**Рис. 7.34.**Диалог Continue with merge

После подтверждения слияния (кнопка OK) модель-источник подсоединяется к модели-цели, *стрелка* вызова исчезает, а *работа*, от которой отходила *стрелка* вызова, становится декомпозируемой — к ней подсоединяется *диаграмма* декомпозиции первого уровня модели-источника. *Стрелки*, касающиеся *работы* на диаграмме модели-цели, автоматически не мигрируют в декомпозицию, а отображаются как неразрешенные. Их следует туннелировать вручную.

В процессе слияния модель-источник остается неизменной, и к модели-цели подключается фактически ее копия. Не нужно путать слияние моделей с синхронизацией. Если в дальнейшем модель-источник будет редактироваться, эти изменения автоматически не попадут в соответствующую *ветвь* модели-цели.

Разделение моделей производится аналогично. Для отщепления ветви от модели следует щелкнуть правой кнопкой мыши по декомпозированной *работе* ( *работа* не должна иметь диагональной черты в левом верхнем углу) и выбрать во всплывающем *меню* *пункт* *Split* *Model*. В появившемся диалоге *Split* Options следует указать имя создаваемой модели. После подтверждения расщепления в старой модели *работа* станет недекомпозированной (признак — диагональная черта в левом верхнем углу), будет создана *стрелка* вызова, ее имя будет совпадать с именем новой модели, и, наконец, будет создана новая модель, причем имя контекстной *работы* будет совпадать с именем *работы*, от которой была "оторвана" *декомпозиция*.

### Создание отчетов в BPwin

BPwin имеет мощный инструмент генерации отчетов. Отчеты по модели вызываются из пункта *меню* *Report*. Всего имеется семь типов отчетов:

1. Model Report. Включает информацию о контексте модели — имя модели, точку зрения, область, цель, имя автора, дату создания и др.
2. Diagram Report. Отчет по конкретной диаграмме. Включает список объектов ( *работ*, стрелок, хранилищ данных, внешних ссылок и т. д.).
3. Diagram Object Report. Наиболее полный отчет по модели. Может включать полный список объектов модели ( *работ*, стрелок с указанием их типа и др.) и свойства, определяемые пользователем.
4. Activity Cost Report. Отчет о результатах стоимостного анализа. Будет рассмотрен ниже.
5. Arrow Report. Отчет по *стрелкам*. Может содержать информацию из словаря стрелок, информацию о работе-источнике, работе-назначении *стрелки* и информацию о разветвлении и слиянии стрелок.
6. Data Usage Report. Отчет о результатах связывания модели процессов и модели данных. (Будет рассмотрен ниже.)
7. Model Consistency Report. Отчет, содержащий список синтаксических ошибок модели.