**3.38. Печать**

Подсистема «Печать» предназначена для формирования печатных форм объектов на основе табличных макетов (формат MXL) или макетов офисных документов в формате Office Open XML (docx).

Подсистема предоставляет инструменты для размещения команд печати на формах в подменю Печать, форму предпросмотра печатных форм, а также различные сервисные возможности по сохранению печатных форм в файлы, отправке по электронной почте и формированию изображений QR-кодов. Кроме того, в подсистему входит рабочее место по редактированию макетов печатных форм.

**Настройка**

Необходимо принять решение по поводу состава объектов конфигурации (справочников, документов и т. п.), которые требуется выводить на печать, и в каком виде должны формироваться печатные формы. Затем создать для них команды печати, разработав описательную часть, логику формирования печатной формы, и внести изменения в модули форм, в которых предполагается выводить команды печати.

Описательная часть находится в процедуре ДобавитьКомандыПечати, а логика формирования печатной формы зависит от того, в каком виде должны формироваться печатные формы:

* ● Формирование печатной формы в формате табличного документа (с предварительным просмотром или сразу на принтер).
* ● Формирование комплекта табличных документов (с предварительным просмотром или сразу на принтер).
* ● Формирование печатных форм с интерактивным запросом дополнительных параметров у пользователя.
* ● Вывод табличного документа в один из популярных форматов (Office Open XML, Microsoft Excel, Adobe PDF,  HTML, текстовый документ и другие).
* ● Формирование печатной формы в виде офисных документов в формате Office Open XML (для тех случаев, когда возможностей табличного макета недостаточно).

**Подключение объектов программы**

Необходимо принять решение, для каких объектов программы разрабатываются команды печати, и в каких их формах требуется выводить подменю с командами печати.

При этом стоит учитывать сценарии подключения команд при помощи расширений. Т.е. если для объекта программы не планируется разрабатывать команды печати в составе конфигурации, но есть вероятность, что партнеры захотят подключить к этому объекту свои команды печати (при помощи расширений), то рекомендуется подключить этот объект.

* ● Объекты конфигурации, являющиеся поставщиками команд печати, следует перечислить в процедуре ПриОпределенииОбъектовСКомандамиПечати модуля УправлениеПечатьюПереопределяемый.
* ● В модуле менеджера каждого из этих объектов, в области ПрограммныйИнтерфейс, должна быть определена процедура ДобавитьКомандыПечати по шаблону:

#Область ПрограммныйИнтерфейс

// Заполняет список команд печати.

//

// Параметры:

//   КомандыПечати - ТаблицаЗначений - состав полей см. в функции УправлениеПечатью.СоздатьКоллекциюКомандПечати.

//

Процедура ДобавитьКомандыПечати(КомандыПечати) Экспорт

КонецПроцедуры

#КонецОбласти

* ● Далее, в формах объектов, подключенных на предыдущем шаге, в которых требуется выводить подменю с командами печати, надо:
* ● Встроить подсистему "[Подключаемые команды](https://its.1c.ru/db/content/bsp301doc/src/3.39.%20подключаемые%20команды.htm?_=1542292879#_поиск_и_удаление)".
* ● Опционально. В командную панель добавить основное подменю для вывода команд отчетов по шаблону:
* ● Имя: "ПодменюПечать",
* ● Заголовок: "Печать",
* ● Вид: Подменю,
* ● Отображение: Картинка,
* ● Картинка: Печать (стандартная картинка).
* Если предполагается что в этом подменю может быть выведено большое количество команд (более 10), то рекомендуется добавить вложенные группы кнопок с суффиксами "Важное", "Обычное" и "СмТакже". Например: "ПодменюПечатьВажное", "ПодменюПечатьОбычное" и "ПодменюПечатьСмТакже". Суффиксы этих групп указываются в свойстве Важность тех команд, которые должны выводиться в этих группах (подробнее про это свойство см. далее в таблице "Параметры команд печати").
* ● В формах журналов документов подменю Печать по умолчанию заполняется командами документов, входящих в журнал. В случаях, когда команды печати необходимо брать непосредственно в модуле менеджера журнала документов, необходимо сделать соответствующую вставку кода в процедуре ПриПолученииНастроекСпискаКомандПечати общего модуля УправлениеПечатьюПереопределяемый:

// Дополнительные настройки списка команд печати.

//

// Параметры:

//  НастройкиСписка - Структура - модификаторы списка команд печати.

//   \* МенеджерКомандПечати     - МенеджерОбъекта - менеджер объекта, в котором формируется список команд печати;

//   \* АвтоматическоеЗаполнение - Булево - заполнять команды печати из объектов, входящих в состав журнала.

//                                         Значение по умолчанию: Истина.

//

Процедура ПриПолученииНастроекСпискаКомандПечати(НастройкиСписка) Экспорт

Если НастройкиСписка.МенеджерКомандПечати = ЖурналыДокументов.\_ДемоСкладскиеДокументы Тогда

НастройкиСписка.АвтоматическоеЗаполнение = Ложь;

КонецЕсли;

КонецПроцедуры

**Разработка команд печати**

Затем необходимо разработать описательную часть, перечислив команды печати для каждого объекта в процедуре ДобавитьКомандыПечати, указав представление, идентификатор и другие параметры (подробнее см. таблицу ниже). Например:

// Счет на оплату

КомандаПечати = КомандыПечати.Добавить();

КомандаПечати.МенеджерПечати = "Документ.\_ДемоСчетНаОплатуПокупателю";

КомандаПечати.Идентификатор = "СчетЗаказ";

КомандаПечати.Представление = НСтр("ru = 'Счет на оплату'");

КомандаПечати.ПроверкаПроведенияПередПечатью = Истина;

Параметры команды печати:

Таблица 3.72. Параметры команд печати.

| Параметр | Тип | Описание |
| --- | --- | --- |
| Идентификатор (обязательный) | Строка или Массив | Идентификатор команды печати, по которому менеджер печати определяет печатную форму, которую необходимо сформировать. Пример:  КомандаПечати.Идентификатор = "СчетЗаказ";  Для печати нескольких печатных форм можно указывать одновременно несколько их идентификаторов (строкой, через запятую либо массивом строк), например:  КомандаПечати.Идентификатор = "СчетЗаказ,ГарантийноеПисьмо";  Если необходимо задать количество копий печати для печатной формы, то ее идентификатор нужно продублировать столько раз, сколько копий необходимо сформировать. При этом следует учитывать, что порядок следования печатных форм в комплекте будет соответствовать порядку идентификаторов печатных форм, указанных в этом параметре. Пример (2 счета на оплату + 1 гарантийное письмо):  КомандаПечати.Идентификатор = "СчетЗаказ,СчетЗаказ,ГарантийноеПисьмо";  Идентификатор печатной формы может содержать в себе и альтернативный менеджер печати, если он отличается от указанного в параметре МенеджерПечати, например:  КомандаПечати.Идентификатор = "СчетЗаказ,Обработка.\_ДемоПечатнаяФорма.ГарантийноеПисьмо";  В этом примере ГарантийноеПисьмо формируется в менеджере печати Обработка.\_ДемоПечатнаяФорма, а СчетЗаказ – в менеджере печати, указанном в параметре МенеджерПечати |
| Представление (обязательный) | Строка | Представление команды в меню Печать. Пример:  КомандаПечати.Представление = НСтр("ru = 'Счет на оплату'"); |
| МенеджерПечати (необязательный) | Строка | Имя объекта, в модуле менеджера которого располагается процедура Печать, формирующая табличные документы для этой команды. Например:  КомандаПечати.МенеджерПечати = "Документ.\_ДемоСчетНаОплатуПокупателю"; |
| ТипыОбъектовПечати  (необязательный) | Массив | Список типов объектов, для которых предназначена команда печати. Параметр предназначен для команд печати в журналах документов, где требуется проверка передаваемого типа объекта перед вызовом менеджера печати.  Если список не заполнен, то при автоматическом создании списка команд печати в журнале документов он заполняется типом объекта, из которого была импортирована команда печати |
| Обработчик  (необязательный) | Строка | Клиентский обработчик команды, в который необходимо передать управление вместо стандартного обработчика команды печать. Используется, например, когда печатная форма формируется на клиенте.  Формат "<ИмяОбщегоМодуля>.<ИмяПроцедуры>" используется, когда процедура размещена в общем модуле.  Формат "<ИмяПроцедуры>" используется, когда процедура размещена в модуле основной формы отчета или обработки, указанной в МенеджерПечати.  Пример:  КомандаПечати.Обработчик = "\_ДемоСтандартныеПодсистемыКлиент.ПечатьСчетовНаОплатуПокупателю";  Пример обработчика в модуле формы:  // Формирует печатную форму <представление печатной формы>.  //  // Параметры:  //   ПараметрыПечати - Структура - Сведения о печатной форме.  //       \* ОбъектыПечати - Массив - Массив ссылок выбранных объектов.  //       \* Форма - УправляемаяФорма - Форма, из которой вызвана команда печати.  //       \* ДополнительныеПараметры - Структура - Дополнительные параметры печати.  //       Прочие ключи структуры соответствуют колонкам таблицы КомандыПечати,  //       подробнее см. в функции УправлениеПечатью.СоздатьКоллекциюКомандПечати.  //  &НаКлиенте  Функция <ИмяФункции>(ПараметрыПечати) Экспорт  // Обработчик печати.  КонецФункции  Следует иметь в виду, что обработчик вызывается при помощи метода Вычислить, поэтому в качестве обработчика может выступать только функция. При этом возвращаемое значение функции никак в дальнейшем не используется подсистемой. |
| Порядок  (необязательный) | Число | Значение от 1 до 100, указывающее порядок размещения команды по отношению к другим командам. Сортировка команд меню Печать осуществляется сначала по полю Порядок, затем по представлению.  Значение по умолчанию: 50 |
| Картинка  (необязательный) | Картинка | Картинка, которая отображается возле команды в меню Печать. Пример:  КомандаПечати.Картинка = БиблиотекаКартинок.ФорматPDF; |
| СписокФорм  (необязательный) | Строка | Имена форм, в которых должна отображаться команда, через запятую. Если параметр не указан, то команда печати будет отображаться во всех формах объекта, где встроена подсистема «Печать». Пример:  КомандаПечати.СписокФорм = "ФормаДокумента,ФормаСписка" |
| МестоРазмещения  (необязательный) | Строка | Имя командной панели формы, в которую необходимо разместить команду печати. Параметр необходимо использовать только в случае, когда на форме размещается более одного подменю "Печать". В остальных случаях место размещения необходимо задавать в модуле формы при вызове метода УправлениеПечатью.ПриСозданииНаСервере. |
| ЗаголовокФормы  (необязательный) | Строка | Произвольная строка, переопределяющая стандартных заголовок формы «Печать документов». Пример:  КомандаПечати.ЗаголовокФормы = НСтр("ru = 'Настраиваемый комплект'"); |
| ФункциональныеОпции  (необязательный) | Строка | Имена функциональных опций, от которых зависит доступность команды печати, через запятую |
| УсловияВидимости (необязательный) | Массив | Определяет видимость команды в зависимости от контекста.  Для регистрации условий следует использовать процедуру Подключае-мыеКоманды.ДобавитьУсловиеВидимостиКоманды().  Условия объединяются по "И". |
| ПроверкаПроведенияПередПечатью  (необязательный) | Булево | Признак необходимости проверки проведенности документов перед печатью. Если параметр не указан, то проверка проведенности не выполняется. Пример:  КомандаПечати.ПроверкаПроведенияПередПечатью = Истина; |
| СразуНаПринтер  (необязательный) | Булево | Признак необходимости печати документов без предварительного просмотра, сразу на принтер. Если параметр не указан, то при выборе команды печати открывается форма предварительного просмотра «Печать документов». Пример:  КомандаПечати.СразуНаПринтер = Истина; |
| ФорматСохранения  (необязательный) | ТипФайлаТабличногоДокумента | Применяется для быстрого сохранения печатной формы (без дополнительных действий) в различные форматы, отличные от mxl. Если параметр не указан, то формируется обычный mxl. Пример:  КомандаПечати.ФорматСохранения = ТипФайлаТабличногоДокумента.PDF;  При выборе команды печати сразу открывается сформированный в формате pdf документ |
| ПереопределитьПользовательскиеНастройкиКоличества  (необязательный) | Булево | Признак необходимости отключения в форме ПечатьДокументов механизма сохранения/восстановления выбранного пользователем количества экземпляров на печать. Если параметр не указан, то механизм сохранения/восстановления настроек будет работать при открытии формы ПечатьДокументов. Пример:  КомандаПечати.ПереопределитьПользовательскиеНастройкиКоличества = Истина; |
| ДополнитьКомплектВнешнимиПечатнымиФормами  (необязательный) | Булево | Признак необходимости дополнить комплект документов всеми подключенными к объекту внешними печатными формами (подсистема «ДополнительныеОтчетыИОбработки»). Если параметр не указан, внешние печатные формы не добавляются в комплект. Пример:  КомандаПечати.ДополнитьКомплектВнешнимиПечатнымиФормами = Истина; |
| ФиксированныйКомплект  (необязательный) | Булево | Признак необходимости блокировки от изменения пользователем состава комплекта документов. Если параметр не указан, то пользователь сможет исключать отдельные печатные формы из комплекта в форме ПечатьДокументов, а также изменять их количество. Пример:  КомандаПечати.ФиксированныйКомплект = Истина; |
| ДополнительныеПараметры  (необязательный) | Структура | Произвольные параметры для передачи в менеджер печати |
| НеВыполнятьЗаписьВФорме  (необязательный) | Булево | Признак необходимости отключения механизма записи объекта перед выполнением команды печати. Используется в исключительных случаях. Если параметр не указан, производится запись объекта, в случае если в форме объекта установлен признак модифицированности. Пример:  КомандаПечати. НеВыполнятьЗаписьВФорме= Истина; |
| ТребуетсяРасширениеРаботыСФайлами  (необязательный) | Булево | Признак необходимости подключения расширения работы с файлами перед выполнением команды. Если параметр не указан, подключение расширения работы с файлами выполняться не будет |

Разработка команд печати существенно отличается в зависимости от того, в каком виде должна формироваться печатная форма:

* ● Для формирования печатной формы в формате табличного документа или комплекта табличных документов см. раздел «[Разработка процедуры «Печать»](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_разработка_процедуры_«печать»)».
* ● Для вывода табличного документа в один из популярных форматов (Office Open XML, Microsoft Excel, Adobe PDF, HTML, текстовый документ и другие) см. раздел «[Разработка процедуры «Печать»](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_разработка_процедуры_«печать»)».
* ● Для формирования печатных форм с интерактивным запросом дополнительных параметров у пользователя см. раздел «[Формирование печатной формы в клиентском контексте](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_формирование_печатной_формы)».
* ● Для формирования печатной формы в виде офисных документов в формате Office Open XML см. раздел «[Использование макетов в формате офисных документов при формировании печатных форм](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_использование_макетов_в)».

**Разработка процедуры «Печать»**

1. В модуль менеджера печати, указанного в параметре МенеджерПечати, добавить экспортную процедуру Печать:

// Формирует печатные формы.

//

// Параметры:

//  МассивОбъектов - Массив - ссылки на объекты, которые нужно распечатать;

//  ПараметрыПечати - Структура - дополнительные настройки печати;

//  КоллекцияПечатныхФорм - ТаблицаЗначений - сформированные табличные документы (выходной параметр)

//  ОбъектыПечати - СписокЗначений - значение - ссылка на объект;

//                                            представление - имя области, в которой был выведен объект (выходной параметр);

//  ПараметрыВывода - Структура - дополнительные параметры сформированных табличных документов (выходной параметр).

//

Процедура Печать(МассивОбъектов, ПараметрыПечати, КоллекцияПечатныхФорм, ОбъектыПечати, ПараметрыВывода) Экспорт

КонецПроцедуры

2. Разместить в конфигурации макет табличного документа согласно инструкциям раздела «[Размещение макетов печатных форм в конфигурации](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_размещение_макетов_печатных)».

3. Добавить функцию, формирующую печатную форму (табличный документ), например:

Функция ПечатьСчетаЗаказа(МассивОбъектов, ОбъектыПечати)

. . .

Возврат ТабличныйДокумент;

КонецФункции

4. В процедуре Печать разместить код для идентификации требуемой печатной формы и вызова функции по ее формированию (созданную на шаге 3). Например:

НужноПечататьМакет = УправлениеПечатью.НужноПечататьМакет(КоллекцияПечатныхФорм, "СчетЗаказ");

Если НужноПечататьМакет Тогда

УправлениеПечатью.ВывестиТабличныйДокументВКоллекцию(

КоллекцияПечатныхФорм,

"СчетЗаказ",

НСтр("ru = 'Счет на оплату'"),

ПечатьСчетаЗаказа(МассивОбъектов, ОбъектыПечати),

,

"Документ.\_ДемоСчетНаОплатуПокупателю.ПФ\_MXL\_СчетЗаказ");

КонецЕсли;

Параметры процедуры Печать:

Таблица 3.73.

| Параметр | Описание |
| --- | --- |
| МассивОбъектов | Массив ссылок на объекты, для которых нужно сформировать табличные документы |
| ПараметрыПечати | Произвольные параметры, переданные при вызове команды печати |
| КоллекцияПечатныхФорм | Служебная таблица значений. Через нее передаются имена макетов, для которых должны быть сформированы табличные документы. Возвращаются синонимы макетов (представление), сформированные табличные документы с указанием количества экземпляров, картинка для отображения на закладке соответствующего табличного документа в форме печати документов |
| ОбъектыПечати | Список значений, в котором обработчик печати устанавливает соответствие между объектами и именами областей печати табличного документа: значение – Объект, представление – имя области, в которой был выведен объект |
| ПараметрыВывода | Структура с предопределенными ключами, через которую обработчик печати может вернуть параметры:  ● ПараметрыОтправки – структура, содержащая информацию для заполнения письма при отправке печатной формы по электронной почте. Содержит следующие поля (описание см. в общем модуле конфигурации РаботаСПочтовымиСообщениямиКлиент в процедуре ОткрытьФормуОтправкиПочтовогоСообщения):  ● Получатель,  ● Тема,  ● Текст |

Процедура Печать может формировать один или несколько табличных документов за один вызов. Процедура может быть реализована следующим образом:

* ● При необходимости устанавливаются ПараметрыВывода.
* ● Если формируется больше одного табличного документа, то рекомендуется получить необходимые данные до формирования табличных документов.
* ● При формировании табличного документа для нескольких объектов необходимо позаботиться о правильном порядке этих объектов в табличном документе. В процедуру Печать в параметр МассивОбъектов передается массив объектов в том порядке, в котором их выбрал пользователь, однако «правильный» порядок вывода может быть иным – например, в документах, как правило, упорядочивание выполняется по полю МоментВремени.
* ● Если для каждого табличного документа данные получаются разными запросами, то желательно, чтобы документы упорядочивались одинаково.

В связи с тем, что макет может быть отредактирован пользователем в режиме 1С:Предприятие и параметры в нем могут быть изменены или удалены, для повышения устойчивости кода формирования печатной формы рекомендуется избегать явного присвоения значений параметров в областях печати. Вместо этого следует использовать процедуру ЗаполнитьЗначенияСвойств.

Неправильно:

ОбластьПечати.Параметры.Организация = ДанныеПечати.Организация;

ОбластьПечати.Параметры.Контрагент = ДанныеПечати.Контрагент;

Правильно:

ЗаполнитьЗначенияСвойств(ОбластьПечати.Параметры, ДанныеПечати);

Для каждого макета, для которого в обработчике предусмотрено формирование табличного документа, необходимо проверить, нужно ли формировать табличный документ. Проверка осуществляется вызовом функции УправлениеПечатью.НужноПечататьМакет с параметрами.

Сформированный табличный документ добавляется в коллекцию вызовом процедуры УправлениеПечатью.ВывестиТабличныйДокументВКоллекцию с параметрами.

Пример реализации процедуры Печать:

Процедура Печать(МассивОбъектов, ПараметрыПечати, КоллекцияПечатныхФорм, ОбъектыПечати, ПараметрыВывода) Экспорт

// Проверяем, нужно ли для макета СчетЗаказ формировать табличный документ.

Если УправлениеПечатью.НужноПечататьМакет(КоллекцияПечатныхФорм, "СчетЗаказ") Тогда

// Формируем табличный документ и добавляем его в коллекцию печатных форм.

УправлениеПечатью.ВывестиТабличныйДокументВКоллекцию(КоллекцияПечатныхФорм,

"СчетЗаказ", "Счет заказ", ПечатьСчетаЗаказ(МассивОбъектов, ОбъектыПечати));

КонецЕсли;

КонецПроцедуры

Пример реализации функции, формирующей табличный документ:

Функция ПечатьСчетаЗаказ(МассивОбъектов, ОбъектыПечати)

// Создаем табличный документ и устанавливаем имя параметров печати.

ТабличныйДокумент = Новый ТабличныйДокумент;

ТабличныйДокумент.КлючПараметровПечати = "ПараметрыПечати\_СчетЗаказ";

// Получаем запросом необходимые данные.

Запрос = Новый Запрос();

Запрос.Текст =

"ВЫБРАТЬ

| СчетНаОплатуПокупателю.Ссылка КАК Ссылка,

......................

|ГДЕ

| СчетНаОплатуПокупателю.Ссылка В(&МассивОбъектов)

......................

|";

Запрос.УстановитьПараметр("МассивОбъектов", МассивОбъектов);

Шапка = Запрос.Выполнить().Выбрать();

ПервыйДокумент = Истина;

Пока Шапка.Следующий() Цикл

Если Не ПервыйДокумент Тогда

// Все документы нужно выводить на разных страницах.

ТабличныйДокумент.ВывестиГоризонтальныйРазделительСтраниц();

КонецЕсли;

ПервыйДокумент = Ложь;

// Запомним номер строки, с которой начали выводить текущий документ.

НомерСтрокиНачало = ТабличныйДокумент.ВысотаТаблицы + 1;

......................

// В табличном документе необходимо задать имя области, в которую был

// выведен объект. Нужно для возможности печати комплектов документов.

УправлениеПечатью.ЗадатьОбластьПечатиДокумента(ТабличныйДокумент,

НомерСтрокиНачало, ОбъектыПечати, Шапка.Ссылка);

КонецЦикла;

Возврат ТабличныйДокумент;

КонецФункции

При разработке печатной формы можно предусмотреть ограничение ее вывода (на принтер, в буфер обмена, сохранение в файл). Для этого при формировании табличного документа необходимо установить у него соответствующее значение в свойстве Вывод, например:

ТабличныйДокумент.Вывод = ИспользованиеВывода.Запретить

**Формирование печатной формы в клиентском контексте**

В отдельных случаях для формирования некоторых печатных форм может потребоваться клиентский контекст. Например, для запроса дополнительных параметров печатной формы у пользователя непосредственно перед печатью. В таких случаях механизм формирования печатной формы необходимо размещать в клиентском модуле, а при описании команды печати в процедуре ДобавитьКомандыПечати использовать параметр Обработчик для передачи управления в этот модуль.

Принцип создания команд, использующих клиентский контекст, несколько отличается от основного. Процедура Печать модуля менеджера объекта не вызывается механизмами подсистемы, поэтому создавать ее для такой команды не требуется.

Процесс создания такой команды выглядит следующим образом:

1. В модуле менеджера объекта в процедуре ДобавитьКомандыПечати добавить описание команды (с использованием параметра Обработчик), например:

// Счет на оплату в Microsoft Word

КомандаПечати = КомандыПечати.Добавить();

КомандаПечати.МенеджерПечати = "Документ.\_ДемоСчетНаОплатуПокупателю";

КомандаПечати.Идентификатор = "СчетНаОплату(MSWord)";

КомандаПечати.Представление = НСтр("ru = 'Счет на оплату в Microsoft Word'");

КомандаПечати.Картинка = БиблиотекаКартинок.ФорматWord2007;

КомандаПечати.ПроверкаПроведенияПередПечатью = Истина;

КомандаПечати.Обработчик = "\_ДемоСтандартныеПодсистемыКлиент.ПечатьСчетовНаОплатуПокупателю";

Примечание

Менеджер печати может быть использован для получения данных на печать (см. раздел «[Использование макетов в формате офисных документов при формировании печатных форм](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_использование_макетов_в)»).

2. Добавить клиентскую экспортную функцию формирования печатной формы с единственным параметром, в который подсистема «Печать» будет передавать структуру параметров команды. Имя функции может быть произвольным, например:

Функция ПечатьСчетовНаОплатуПокупателю(ПараметрыПечати) Экспорт

.

.

.

КонецФункции

Примечание 1

Из этой функции подсистема «Печать» не ожидает никакого результата. Использование функции вместо процедуры обусловлено тем, что ее вызов осуществляется при помощи метода Вычислить().

Примечание 2

Для вывода печатных форм на принтер пользователь должен обладать правом «Вывод». При разработке клиентских команд печати следует учитывать наличие этого права.

Если клиентский контекст предполагает только запрос дополнительных параметров, а формирование табличного документа выполняется на сервере, то необходимо также выполнить [инструкцию по разработке процедуры Печать](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_разработка_процедуры_«печать»), а из клиентской функции выполнять передачу управления в процедуру УправлениеПечатьюКлиент.ВыполнитьКомандуПечати.

**Вывод в печатную форму изображения QR-кода по заданной текстовой строке**

В некоторых случаях в печатной форме документа полезно разместить изображение QR-кода, содержащего закодированные данные. Общий принцип встраивания функционала следующий:

1. В макете печатной формы разместить рисунок типа Картинка.

2. В модуле менеджера печати вызвать интерфейсный метод УправлениеПечатью.ДанныеQRКода, передав в качестве параметра текстовую строку, которую необходимо закодировать с помощью QR-кода.

3. Полученный результат (двоичные данные) программно вывести в рисунок типа Картинка.

Пример реализации см. в демонстрационной конфигурации, в модуле менеджера документа \_ДемоСчетНаОплатуПокупателю в процедуре ПечатьКвитанции и макете ПФ\_MXL\_Квитанция.

**Размещение макетов печатных форм в конфигурации**

Макеты печатных форм рекомендуется размещать при объекте, в котором реализована логика формирования печатных форм. К имени макета следует добавить специальный префикс:

* ● макеты табличных документов – ПФ\_MXL,
* ● макет Office Open XML – ПФ\_DOC,

Например, ПФ\_DOC\_СчетНаОплату.

В процедурах формирования печатных форм для получения макета следует использовать процедуру УправлениеПечатью.ПолучитьМакет(<Путь к макету>), где путь к макету может быть:

* ● для макетов в документах: Документ.<Имя документа>.<Имя макетов>;
* ● для макетов в обработках: Обработка.<Имя обработки>.<Имя макета>;
* ● для общих макетов: ОбщийМакет.<Имя макета>.

Например:

УправлениеПечатью.ПолучитьМакет("Документ.СчетНаОплатуПокупателю.ПФ\_MXL\_СчетЗаказ");

**Разработка печатных форм с использованием макетов в формате офисных документов Office Open XML**

Помимо макетов в формате табличного документа (mxl), можно использовать макеты в формате Office Open XML (docx).

В общем виде разработка такой печатной формы выглядит следующим образом:

1. Создание макета печатной формы.

2. Создание процедуры подготовки данных для печати.

3. Создание описания команды печати.

4. Разработка обработчика команды печати.

**Создание макета печатной формы**

Макет печатной формы может быть подготовлен в любых офисных пакетах, работающих со стандартом Office Open XML: Microsoft Word, OpenOffice.org Writer, Google Docs и др. Макет представляет собой обычный офисный документ, состоящий из одной или нескольких областей следующих типов:

* ● СтрокаТаблицы – используется для вывода множества строк;
* ● Список – используется для вывода множества строк;
* ● Общая – используется однократно.

Области выделяются при помощи разделителей специального вида:

{v8 Область.<Имя области>}

...

{/v8 Область.<Имя области>}

Каждая область может содержать в себе маркеры параметров для подстановки значений из информационной базы:

{v8 <Имя параметра>}

Макет необходимо разместить в конфигурации (см. «[Размещение макетов печатных форм в конфигурации](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_размещение_макетов_печатных)»).

**Создание процедуры подготовки данных для печати**

В модуле менеджера объекта необходимо добавить экспортную процедуру ПолучитьДанныеПечати со следующими параметрами:

Таблица 3.74.

|  |  |
| --- | --- |
| Параметр функции | Описание |
| МассивДокументов | Массив ссылок на объекты, для которых запрашиваются данные для печати |
| МассивИменМакетов | Массив имен макетов, в которые подставляются данные для печати |
| ДополнительныеПараметры | Структура с дополнительными параметрами |

Возвращаемое значение функции – соответствие с двумя ключами: Данные и Макеты. В ключе Данные находится соответствие в формате:

* ● ключ – ссылка на объект информационной базы;
* ● значение – структура, где ключ – имя макета, а значение – структура с данными объекта (формат выбирается самостоятельно).

В ключе Макеты находится структура с тремя ключами: ОписаниеОбластей, ТипыМакетов и ДвоичныеДанныеМакетов. В ключе ОписаниеОбластей содержится структура, в которой в качестве ключа выступает имя макета, а значением является структура, содержащая описание области макета, формируемая с помощью вспомогательной функции ДобавитьОписаниеОбласти.



В ключе ДвоичныеДанныеМакетов находится соответствие, в котором ключом является имя макета, а значение содержит двоичные данные макета офисного документа.

**Создание описания команды печати**

В модуле менеджера объекта в процедуре ДобавитьКомандыПечати добавить описание команды, например:

// Счет на оплату в Office Open XML

КомандаПечати = КомандыПечати.Добавить();

КомандаПечати.МенеджерПечати = "Документ.\_ДемоСчетНаОплатуПокупателю";

КомандаПечати.Идентификатор = "СчетНаОплату(OfficeOpenXML)";

КомандаПечати.Представление = НСтр("ru = 'Счет на оплату в Office Open XML'");

КомандаПечати.Картинка = БиблиотекаКартинок.ФорматWord2007;

КомандаПечати.ПроверкаПроведенияПередПечатью = Истина;

КомандаПечати.ТребуетсяРасширениеРаботыСФайлами = Истина;

Описание параметров процедуры ДобавитьКомандыПечати см. в разделе «[Разработка команд печати](https://its.1c.ru/db/content/bsp301doc/src/3.38.%20%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D1%8C.htm?_=1542292879#_разработка_команд_печати)».

**Разработка обработчика команды печати**

Код формирования печатной формы размещается в модуле менеджера печати в процедуре Печать, как и для печатных форм в формате табличного документа. В нем следует использовать программный интерфейс общего модуля УправлениеПечатью из области РаботаСМакетамиОфисныхДокументов.

Пример реализации см. в демонстрационной конфигурации в модуле менеджера документа \_ДемоСчетНаОплатуПокупателю в процедурах ДобавитьКомандыПечати (см. команду Счет на оплату в Office Open XML), ПолучитьДанныеПечати, Печать и в функции НапечататьСчетНаОплатуПокупателю.

Для макетов, в которых определены разделы и колонтитулы, также доступны следующие типы областей:

* ● ВерхнийКолонтитул
* ● НижнийКолонтитул
* ● ВерхнийТитульныйКолонтитул
* ● НижнийТитульныйКолонтитул
* ● ВерхнийЧетныйКолонтитул
* ● НижнийЧетныйКолонтитул

Имя областей формируется по правилу ТипОбласти\_НомерРаздела, где НомерРаздела - порядковый номер раздела в документе, к которому относится колонтитул. Номер раздела можно не указывать, тогда по умолчанию будет считаться, что это первый раздел. Пример:

Область = УправлениеПечатью.ОбластьМакета(Макет, Области[ИмяМакета]["ВерхнийЧетныйКолонтитул\_3"]);

УправлениеПечатью.ПрисоединитьОбластьИЗаполнитьПараметры(ПечатнаяФорма, Область, ДанныеОбъекта);

Кроме того, для вывода в печатную форму абзаца без текста и форматирования можно указывать предопределенную область вывода абзаца Абзац. Например:

Область = УправлениеПечатью.ОбластьМакета(Макет, Области[ИмяМакета]["Абзац"]);

УправлениеПечатью.ПрисоединитьОбласть(ПечатнаяФорма, Область);

**Размещение в командном интерфейсе**

1. Для редактирования макетов печатных форм следует разместить в командном интерфейсе ответственного за изменение макетов регистр сведений ПользовательскиеМакетыПечати.

2. В персональных настройках пользователя следует разместить:

* ● Команду настройки каталога печати с кодом:

ОткрытьФорму("РегистрСведений.ПользовательскиеМакетыПечати.Форма.НастройкаКаталогаФайловПечати");

* ● Команду для задания действия при выборе макета:

ОткрытьФорму("РегистрСведений.ПользовательскиеМакетыПечати.Форма.ВыбораРежимаОткрытияМакета");

**Настройка прав доступа пользователей**

Для настройки прав доступа пользователей к данным подсистемы «Печать» следует использовать роли, указанные ниже.

Таблица 3.75.

| № | Роли и их назначение |
| --- | --- |
| 1. | ИзменениеМакетовПечатныхФорм  Редактирование макетов печатных форм |
| 2. | ВыводНаПринтерФайлБуферОбмена (из подсистемы «Базовая функциональность»)  Используется для разрешения вывода печатных форм на принтер и отправки их по электронной почте |
| 3. | РедактированиеПечатныхФорм  Внесение изменений в сформированные печатные формы перед непосредственной отправкой на принтер, сохранением в файл или по электронной почте. |

Дополнительно следует создать вспомогательные роли или использовать подходящие роли, существующие в конфигурации, для обеспечения доступа к данным.

Таблица 3.76.

|  |  |
| --- | --- |
| № | Вспомогательные роли и их назначение |
| 1. | <ВыводПечатнойФормыНаПринтер>  Просмотр команды вывода печатных форм на принтер |

Пример настройки прав доступа пользователей:

Таблица 3.77.

| № | Группа пользователей и ее функции | Состав ролей |
| --- | --- | --- |
| 1. | Ответственный за поддержку макетов печатных форм | ● БазовыеПраваБСП (из подсистемы «Базовая функциональность»),  ● ЗапускТонкогоКлиента (из подсистемы «Базовая функциональность»),  ● ИзменениеМакетовПечатныхФорм |
| 2. | Пользователь, которому разрешено получать и формировать печатные формы, а также выводить их на печать и пользоваться командами «быстрой печати» на принтер | ● БазовыеПраваБСП или БазовыеПраваВнешнихПользователейБСП (из подсистемы «Базовая функциональность»),  ● ЗапускТонкогоКлиента (из подсистемы «Базовая функциональность»),  ● ВыводНаПринтерФайлБуферОбмена (из подсистемы «Базовая функциональность»),  ● <ВыводПечатнойФормыНаПринтер> |
| 3. | Пользователь, которому разрешено получать и формировать печатные формы, выводить их на печать,  пользоваться командами «быстрой печати» на принтер, а также редактировать, сохранять и оправлять по почте печатные формы | ● БазовыеПраваБСП или БазовыеПраваВнешнихПользователейБСП (из подсистемы «Базовая функциональность»),  ● ЗапускТонкогоКлиента (из подсистемы «Базовая функциональность»),  ● ВыводНаПринтерФайлБуферОбмена (из подсистемы «Базовая функциональность»),  ● РедактированиеПечатныхФорм  ● <ВыводПечатнойФормыНаПринтер> |

**Использование при разработке конфигурации**

**Перенос макета в другой объект**

Обычно макет печатной формы располагается при объекте, но, если макет является общим для двух и более объектов, его следует располагать либо в обработке, либо это должен быть общий макет конфигурации. Таким образом, в процессе разработки конфигурации может возникнуть необходимость переместить макет из одного места конфигурации в другое. При этом следует помнить о том, что пользователь в режиме предприятия мог вносить изменения в макет, и эти изменения, хранящиеся в регистре сведений ПользовательскиеМакетыПечати также необходимо переместить по новому адресу. Для этого необходимо написать отложенный обработчик обновления и воспользоваться в нем программным интерфейсом в модуле УправлениеПечатью:

// Используется при переносе макета (объекта метаданных) печатной формы в другой объект.

// Предназначена для вызова в процедуре заполнения данных обновления (для "отложенного" обработчика).

// Регистрирует новый адрес макета для обработки.

//

// Параметры:

//  ИмяМакета   - Строка - Новое имя макета в формате

//                         "Документ.<ИмяДокумента>.<ИмяМакета>"

//                         "Обработка.<ИмяОбработки>.<ИмяМакета>"

//                         "ОбщийМакет.<ИмяМакета>":

//  Параметры - Структура - см. ОбновлениеИнформационнойБазы.ОсновныеПараметрыОтметкиКОбработке.

//

Процедура ЗарегистрироватьНовоеИмяМакета(ИмяМакета, Параметры) Экспорт

// Используется при переносе макета (объекта метаданных) печатной формы в другой объект.

// Предназначена для вызова в "отложенном" обработчике обновления.

// Переносит пользовательские данные, относящиеся макету на новый адрес.

//

// Параметры:

//  Макеты     - Соответствие - Сведения о прежних и новых именах макетов в формате

//                              "Документ.<ИмяДокумента>.<ИмяМакета>"

//                              "Обработка.<ИмяОбработки>.<ИмяМакета>"

//                              "ОбщийМакет.<ИмяМакета>":

//   \* Ключ     - Строка - Новое имя макета.

//   \* Значение - Строка - Прежнее имя макета.

//

//  Параметры - Структура - параметры, передаваемые в "отложенный" обработчик обновления.

//

Процедура ПеренестиПользовательскиеМакеты(Макеты, Параметры) Экспорт

Например, требуется перенести пользовательский макет (и переименовать)

из "Документ.\_ДемоСчетНаОплатуПокупателю.ПФ\_MXL\_СчетЗаказ"

в "Обработка.\_ДемоПечатьЗаказов.ПФ\_MXL\_СчетУниверсальный"

Процедура ПриДобавленииОбработчиковОбновления(Обработчики) Экспорт

Обработчик = Обработчики.Добавить();

Обработчик.Версия = "2.3.5.51";

Обработчик.Процедура = "\_ДемоОбновлениеИнформационнойБазыБСП.ПеренестиПользовательскиеМакетыВНовыеОбъекты";

Обработчик.РежимВыполнения = "Отложенно";

Обработчик.Комментарий = НСтр("ru = 'Изменение места хранения пользовательских макетов'");

Обработчик.Идентификатор = Новый УникальныйИдентификатор("4c580abd-475c-4e93-a8e5-58219102f660");

Обработчик.ПроцедураПроверки = "ОбновлениеИнформационнойБазы.ДанныеОбновленыНаНовуюВерсиюПрограммы";

Обработчик.БлокируемыеОбъекты = "РегистрСведений.ПользовательскиеМакетыПечати";

Обработчик.ПроцедураЗаполненияДанныхОбновления = "\_ДемоОбновлениеИнформационнойБазыБСП.ЗарегистрироватьДанныеКОбработкеДляПереходаНаНовуюВерсию";

Обработчик.ОчередьОтложеннойОбработки = 1;

Обработчик.ЧитаемыеОбъекты = "РегистрСведений.ПользовательскиеМакетыПечати";

Обработчик.ИзменяемыеОбъекты = "РегистрСведений.ПользовательскиеМакетыПечати";

КонецПроцедуры

Процедура ЗарегистрироватьДанныеКОбработкеДляПереходаНаНовуюВерсию(Параметры) Экспорт

УправлениеПечатью.ЗарегистрироватьНовоеИмяМакета("Обработка.\_ДемоПечатьЗаказов.ПФ\_MXL\_СчетУниверсальный", Параметры);

КонецПроцедуры

Процедура ПеренестиПользовательскиеМакетыВНовыеОбъекты(Параметры) Экспорт

Макеты = Новый Соответствие;

Макеты.Вставить("Обработка.\_ДемоПечатьЗаказов.ПФ\_MXL\_СчетУниверсальный", "Документ.\_ДемоСчетНаОплатуПокупателю.ПФ\_MXL\_СчетЗаказ");

УправлениеПечатью.ПеренестиПользовательскиеМакеты(Макеты, Параметры);

КонецПроцедуры

**Разработка команд печати в расширении конфигурации**

Вместо поставки внешних печатных форм в виде внешних обработок рекомендуется вести их разработку с помощью расширений конфигурации. Для поставки команд печати внешних печатных форм в расширении конфигурации необходимо:

1. Добавить в расширение конфигурации обработку и включить ее в состав подсистемы ПодключаемыеОтчетыИОбработки.

2. В модуле менеджера обработки определить процедуру ПриОпределенииНастроек в соответствии с шаблоном [подключения отчетов и обработок к другим объектам метаданных](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_подключение_отчетов_и) и реализовать команды печати:

#Область ПрограммныйИнтерфейс

// Определяет состав программного интерфейса для интеграции с конфигурацией.

//

// Параметры:

//   Настройки - Структура - Настройки интеграции этого объекта.

//       См. возвращаемое значение функции ПодключаемыеКоманды.НастройкиПодключаемыхОтчетовИОбработок().

//

Процедура ПриОпределенииНастроек(Настройки) Экспорт

Настройки.Размещение.Добавить(Метаданные.Документы.ИмяДокумента);

Настройки.ДобавитьКомандыПечати = Истина;

КонецПроцедуры

// Заполняет список команд печати.

//

// Параметры:

//   КомандыПечати - ТаблицаЗначений - Подробнее см. в УправлениеПечатью.СоздатьКоллекциюКомандПечати().

//

Процедура ДобавитьКомандыПечати(КомандыПечати) Экспорт

// Код по добавлению команд печати.

КонецПроцедуры

#КонецОбласти

// Формирует печатные формы.

//

// Параметры:

//  МассивОбъектов - Массив - ссылки на объекты, которые нужно распечатать;

//  ПараметрыПечати - Структура - дополнительные настройки печати;

//  КоллекцияПечатныхФорм - ТаблицаЗначений - сформированные табличные документы (выходной параметр)

//  ОбъектыПечати - СписокЗначений - значение - ссылка на объект;

//                                            представление - имя области, в которой был выведен объект (выходной параметр);

//  ПараметрыВывода - Структура - дополнительные параметры сформированных табличных документов (выходной параметр).

//

Процедура Печать(МассивОбъектов, ПараметрыПечати, КоллекцияПечатныхФорм, ОбъектыПечати, ПараметрыВывода) Экспорт

КонецПроцедуры

Аналогичным образом в расширение конфигурации можно включить сразу несколько обработок с внешними печатными формами. Кроме того, предусмотрена возможность разрабатывать команды печати в отчетах. Например, это может потребоваться, если для формирования печатной формы используется система компоновки данных (СКД).

См. примеры в демонстрационной конфигурации: обработка \_ДемоПодключаемыеКомандыПечатьСчетовНаОплатуПокупателю в составе расширения \_ДемоПодключаемыеКоманды и обработка \_ДемоПечатьКарточкиОрганизации в составе конфигурации.

При необходимости, разработанные таким образом обработки также можно перенести без каких-либо доработок из расширения конфигурации в состав самой конфигурации.

**Настройка обмена данными**

Все объекты метаданных подсистемы, содержащие данные, рекомендуется включать в планы обмена распределенной ИБ (РИБ).