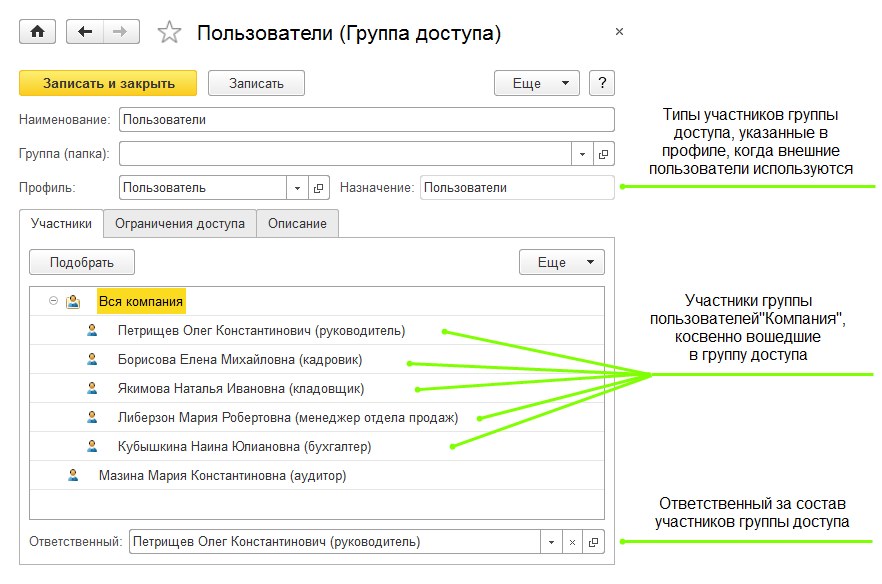
**3.58. Управление доступом**

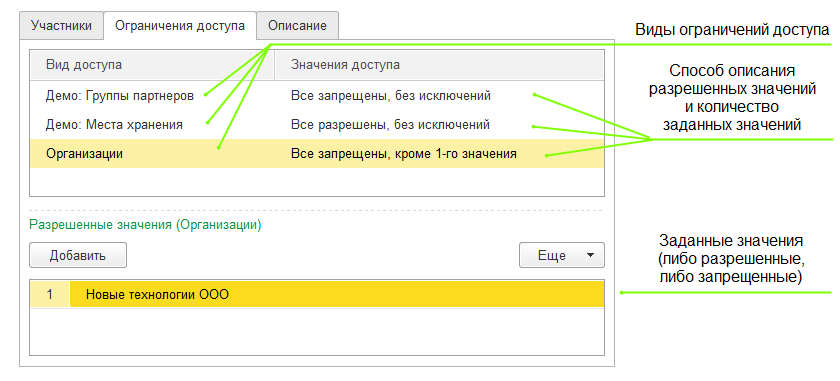
Подсистема «Управление доступом» позволяет настраивать права пользователей для произвольных элементов данных информационной базы (элементов справочников, документов, записей регистров, бизнес-процессов, задач и т. д.). Возможно ограничение прав как для отдельных типов объектов метаданных, так и на уровне записей одного типа объекта. Кроме того, для отдельных объектов информационной базы возможна индивидуальная настройка прав доступа подобно папкам файлов операционной системы.

Для своей работы подсистема использует возможности подсистемы «Пользователи».

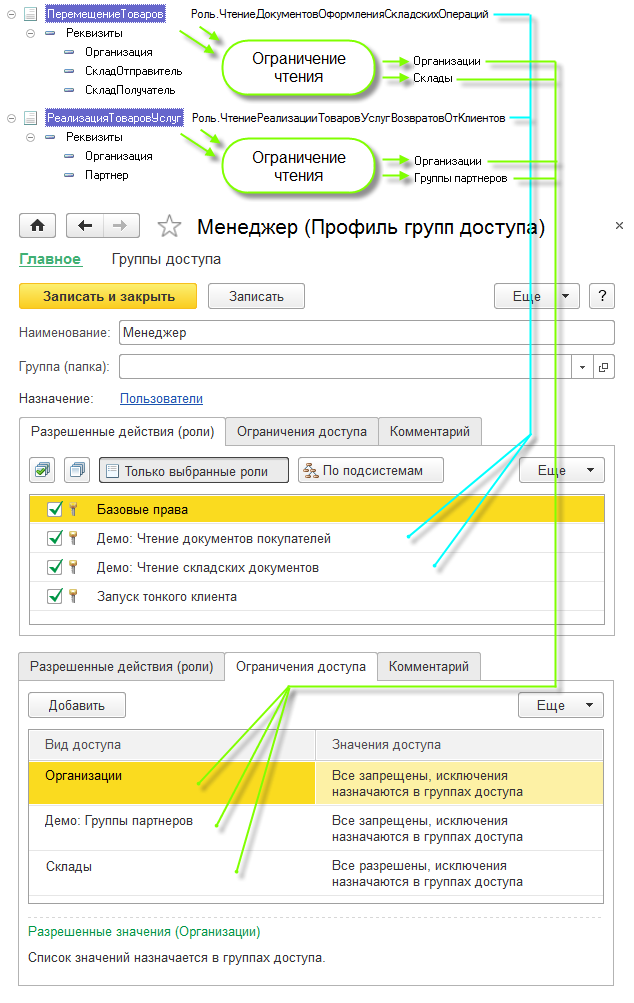
**Группы доступа и их профили**

Для настройки прав и ограничений прав пользователей и групп пользователей предназначен справочник Группы доступа. Если для некоторых объектов информационной базы разрешен доступ с помощью внешних пользователей, то в группы доступа можно также помещать внешних пользователей и группы внешних пользователей.





Состав и логика ограничения прав доступа определяются профилем, заданным в группе доступа.



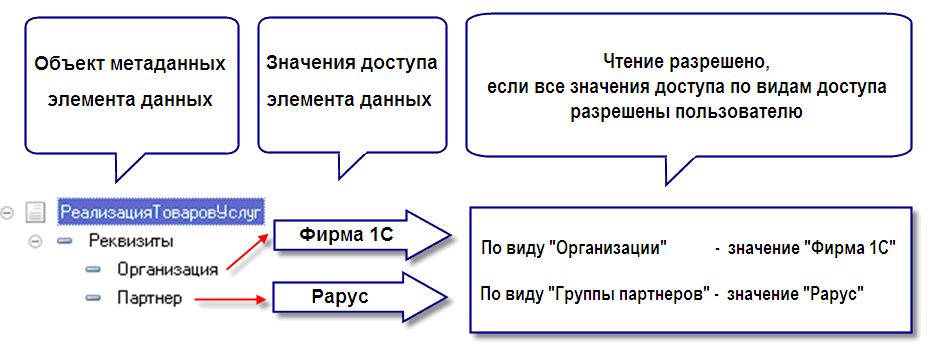
Профиль групп доступа представляет собой совокупность ролей (разрешенных действий) и видов доступа (в разрезе которых ограничивается доступ к данным информационной базы).

Вид доступа – это тип (или несколько типов) объектов информационной базы, в разрезе которых ограничиваются права доступа к данным информационной базы. Например, право чтения документов информационной базы может быть ограничено в разрезе определенных организаций, разрешенных для пользователя. В простейшем случае, если право чтения ограничено по виду доступа Организации, то будут прочитаны только те элементы данных (документы), у которых в поле Организация указана организация из списка разрешенных (или незапрещенных) организаций. Если же в группе доступа ограничение по виду доступа Организации не задано, то могут быть прочитаны все документы (как будто задан пустой список запрещенных организаций).

В самой группе доступа настраиваются списки разрешенных или запрещенных значений по видам доступа, например: списки организаций, групп партнеров, складов, групп номенклатуры и т. д., которые используются в логике ограничения прав для объектов информационной базы.

Таким образом, можно сказать, что у отдельного пользователя есть право доступа к объекту информационной базы, если пользователю разрешены связанные с этим объектом значения видов доступа.

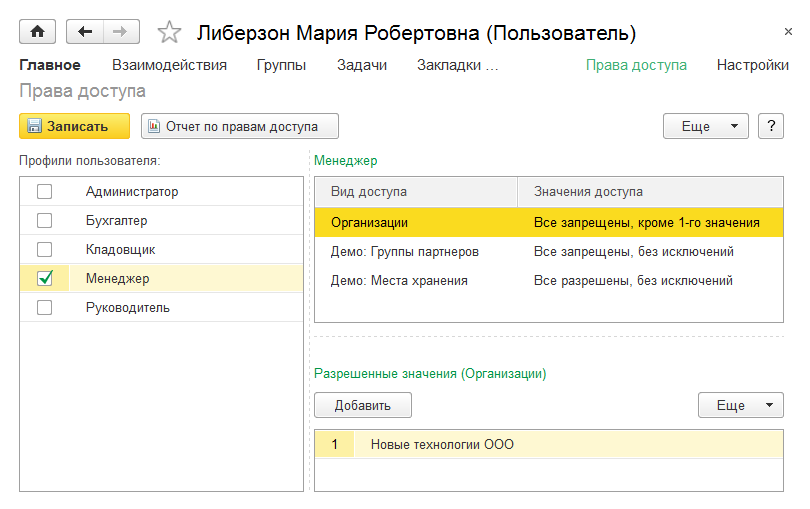
См. также раздел «[Разработка ролей и профилей групп доступа](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_разработка_ролей_и)».



**Упрощенный интерфейс настройки прав доступа**

Для простых конфигураций предусмотрен режим упрощенного интерфейса (включается в переопределяемом модуле).

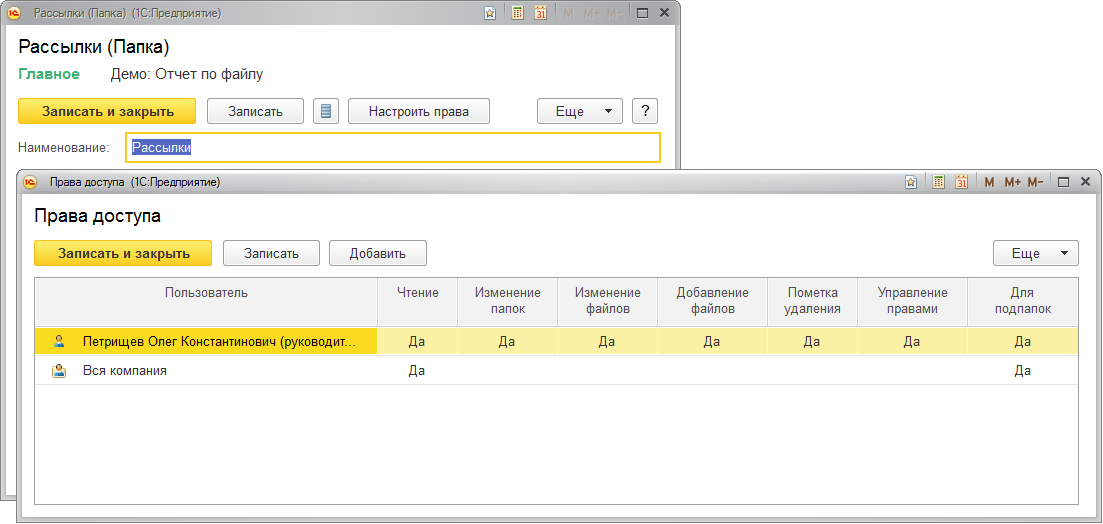
В этом режиме пользователю ставится в соответствие один или несколько профилей, по каждому из которых можно настроить предусмотренные ограничения прав доступа.



**Настройка прав доступа к отдельным объектам информационной базы**

Для отдельных объектов информационной базы возможна индивидуальная настройка прав доступа подобно папкам файлов операционной системы.

Например, для объектов справочника Папки файлов можно настроить доступ пользователей к отдельным папкам, а также содержащимся в них дочерним папкам и элементам справочника Файлы. При этом индивидуальная настройка прав доступа к папкам возможна только для тех пользователей, которые входят в группу доступа пользователей со специальной ролью Работа с папками файлов.



Пример настройки групп доступа пользователей:

1. Профиль Работа с папками файлов (дополнительно) содержит роль Работа с папками файлов, которая предоставляет доступ к справочникам Папки файлов и Файлы.

2. Группа доступа Работа с папками файлов (дополнительно) содержит профиль Работа с папками файлов (дополнительно).

Рекомендуется использовать одну группу доступа для всех пользователей. Тогда сочетание с другими группами доступа будет просто представить, т. к. в группе будут все права на папки, файлы и другие списки, доступ к которым зависит от настроек прав на папки файлов.

**Настройка**

Включает в себя последовательность действий:

1. [Выбор режима обычного или упрощенного интерфейса](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_выбор_режима_обычного).

2. [Определить состав ролей и профилей групп доступа в конфигурации](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_определить_состав_ролей).

3. [Описать логику ограничений прав и составить список требуемых видов доступа](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_описать_логику_ограничений).

4. [Разработка видов доступа. Добавление дополнительных метаданных и размещение некоторых из них на формах интерфейса пользователя](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_разработка_видов_доступа.).

5. [Разработка прав для настройки прав отдельных объектов](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_разработка_прав_для_1).

6. [Настройка вычисления использования видов доступа в зависимости от настроек программы](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_настройка_вычисления_использования).

7. [Разработка ролей профилей групп доступа](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_разработка_ролей_и).

8. [Разработка ограничений прав доступа](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_разработка_ограничений_прав).

9. [Разработка процедур «Заполнить наборы значений доступа»](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_разработка_процедур_«заполнить).

10. [Настройка свойств объектов метаданных](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_настройка_свойств_объектов).

11. [Описание видов доступа, используемых в ограничениях объектов для отчета «Права доступа»](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_описание_видов_доступа,).

12. [Создание описаний поставляемых профилей групп доступа для начального заполнения и обновления информационной базы](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_создание_описаний_поставляемых).

13. [Размещение в командном интерфейсе](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_размещение_в_командном).

14. [Настройка прав доступа пользователей](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_настройка_прав_доступа).

**Выбор режима обычного или упрощенного интерфейса**

Для простых конфигураций с небольшим числом пользователей рекомендуется использовать упрощенный интерфейс. В режиме упрощенного интерфейса блокируется изменение поставляемых профилей (создавать новые допустимо).

Для включения упрощенного интерфейса следует, во-первых, изменить возвращаемое значение в переопределяемом модуле.

Процедура ПриОпределенииИнтерфейсаНастройкиДоступа(УпрощенныйИнтерфейс) Экспорт

УпрощенныйИнтерфейс = Истина;

КонецПроцедуры

Во-вторых, следует изменить в общей команде ПраваДоступа свойство Группа с Панель навигации формы.Перейти на Командная панель формы.Важное.

Не следует размещать списки профилей и групп доступа в интерфейс администратора (доступ к этим спискам только для нестандартных случаев через «Все функции»).

**Определить состав ролей и профилей групп доступа в конфигурации**

Методические рекомендации по проектированию состава ролей конфигурации изложены в разделе «[Базовая функциональность](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_базовая_функциональность)».

Состав профилей может быть определен, например, исходя из должностных обязанностей сотрудников, работающих в системе.

**Описать логику ограничений прав и составить список требуемых видов доступа**

При проектировании логики ограничений доступа рекомендуется придерживаться следующего подхода.

Для каждого объекта метаданных предлагается сформулировать правила ограничения доступа.

* ● описать правила ограничения чтения;
* ● описать правила ограничения добавления и изменения.

При описании правил следует использовать определенные реквизиты объекта, содержащие ссылки на объекты (организации, контрагенты: далее – значения доступа), которые будут использоваться для отбора разрешенных объектов (например, документов). Следует уточнять, как именно должны работать отборы (далее – логика ограничения): требуется ли проверять, что, например, организация и контрагент разрешены для пользователя одновременно, или же достаточно, что пользователю разрешена либо организация, либо контрагент.

Например, опишем логику ограничения доступа к документу Перемещение товаров, ответив на вопрос «Как нужно ограничить доступ к документу?»:

1. Разрешить чтение документа пользователю, если ему разрешены указанная в документе организация и склад-отправитель или склад-получатель.

2. Разрешить добавление и изменение документа пользователю, если ему разрешены организация и склад-отправитель и склад-получатель.

3. Удаление всегда запрещено, кроме пользователя с ролью ПолныеПрава.

Отсюда следует, что доступ к документу должен ограничиваться в разрезе элементов справочников Организации и Склады. И, следовательно, в список видов доступа надо включить виды доступа Организации и Склады.

В результате мы составим полный список видов доступа для всех объектов метаданных конфигурации.

В то же время по сформулированным выше правилам в дальнейшем будут разработаны тексты ограничения доступа для ролей конфигурации.

При этом следует иметь в виду, что логика ограничения доступа к объектам метаданных всегда описывается независимо от состава ролей конфигурации, которые предоставляют доступ к объектам метаданных. Это требование продиктовано соображением производительности работы системы: при попытке доступа к объекту от лица пользователя, которому назначено несколько таких ролей, платформа 1С:Предприятие проверяет совпадение текстов ограничений у прав доступа в этих ролях и, если совпадения нет, выполняет отдельно 2 запроса, каждый со своим текстом ограничения, а результаты запросов объединяет, что сильно снижает производительность системы. Для того чтобы исключить снижение производительности, при наличии в конфигурации нескольких ролей, предоставляющих доступ к одному объекту метаданных, следует указывать в таких ролях одинаковые тексты ограничений у соответствующих прав доступа.

Все объекты метаданных, для которых описана логика ограничения, необходимо указать в процедуре ПриЗаполненииСписковСОграничениемДоступа общего модуля УправлениеДоступомПереопределяемый.

Для выбранных объектов метаданных необходимо вставить в модуль формы объекта в процедуру обработчика события ПриЧтенииНаСервере вызов:

УправлениеДоступом.ПриЧтенииНаСервере(ЭтотОбъект, ТекущийОбъект);

При выполнении вызова будет проверено право изменения объекта, и если права изменения нет, тогда свойство формы ТолькоПросмотр будет установлено Истина.

**Разработка видов доступа. Добавление дополнительных метаданных и размещение некоторых из них на формах интерфейса пользователя**

Для добавления в конфигурацию нового вида доступа необходимо в модуле УправлениеДоступомПереопределяемый в процедуру ПриЗаполненииВидовДоступа добавить описания видов доступа, а также расширить состав определяемых типов.

Например, для вида доступа МестаХранения указать тип справочник МестаХранения. Если доступ планируется ограничивать не по отдельным значениям вида доступа, а по их группам, то дополнительно следует включить типы объектов-групп. Например, для вида доступа ГруппыНоменклатуры указать тип значений справочник Номенклатура и тип групп значений справочник ГруппаДоступаНоменклатуры.

В некоторых случаях у вида доступа может быть несколько типов значений, тогда для их добавления следует использовать процедуру ДобавитьДополнительныеТипыВидаДоступа модуля УправлениеДоступом.

Может понадобиться для одного значения (например, номенклатуры) указывать не одну группу значений (группу доступа номенклатуры), а несколько. В этом случае следует установить свойство вида доступа НесколькоГруппЗначений в Истина.

Например, код процедуры может выглядеть следующим образом:

// Заполняет виды доступа, используемые в ограничениях прав доступа.

// Виды доступа Пользователи и ВнешниеПользователи уже заполнены.

// Их можно удалить, если они не требуются для ограничения прав доступа.

//

// Параметры:

//  ВидыДоступа - ТаблицаЗначений - с колонками:

//   \* Имя                    - Строка - имя, используемое в описании поставляемых

//                                       профилей групп доступа и текстах ОДД.

//   \* Представление          - Строка - представляет вид доступа в профилях и группах доступа.

//   \* ТипЗначений            - Тип    - тип ссылки значений доступа.       Например,

//                                       Тип("СправочникСсылка.Номенклатура").

//   \* ТипГруппЗначений       - Тип    - тип ссылки групп значений доступа. Например,

//                                       Тип("СправочникСсылка.ГруппыДоступаНоменклатуры").

//   \* НесколькоГруппЗначений - Булево - Истина указывает, что для значения доступа (Номенклатуры) можно

//                                       выбрать несколько групп значений (Групп доступа номенклатуры).

//

Процедура ПриЗаполненииВидовДоступа(ВидыДоступа) Экспорт

ВидДоступа = ВидыДоступа.Добавить();

ВидДоступа.Имя = "МестаХранения";

ВидДоступа.Представление = НСтр("ru = 'Места хранения'");

ВидДоступа.ТипЗначений   = Тип("СправочникСсылка.\_ДемоМестаХранения");

ВидДоступа = ВидыДоступа.Добавить();

ВидДоступа.Имя = "ГруппыНоменклатуры";

ВидДоступа.Представление    = НСтр("ru = 'Группы номенклатуры'");

ВидДоступа.ТипЗначений      = Тип("СправочникСсылка.\_ДемоНоменклатура");

ВидДоступа.ТипГруппЗначений = Тип("СправочникСсылка.\_ДемоГруппыДоступаНоменклатуры");

УправлениеДоступом.ДобавитьДополнительныеТипыВидаДоступа(ВидДоступа,

Тип("СправочникСсылка.\_ДемоВидыНоменклатуры"),

Тип("СправочникСсылка.\_ДемоГруппыДоступаНоменклатуры"));

…

КонецПроцедуры

Если для вида доступа требуется указать тип групп значений, то нужно создать дополнительный справочник (например, ГруппыДоступаНоменклатура). При этом у значения доступа (номенклатуры) нужно создать дополнительный реквизит ГруппаДоступа с типом групп значений (например, ГруппыДоступаНоменклатуры). Если вдобавок свойство НесколькоГруппЗначений установлено Истина, тогда вместо реквизита ГруппаДоступа нужно создать табличную часть ГруппыДоступа с реквизитом ГруппаДоступа того же типа.

Добавленные реквизит ГруппаДоступа или табличную часть ГруппыДоступа с реквизитом ГруппаДоступа нужно разместить в форме элемента значения доступа (например, справочника Номенклатура).

В определяемый тип ЗначениеДоступа нужно добавить типы значений доступа и типы групп значений доступа. В определяемый тип ЗначениеДоступаСГруппамиЗначенийДоступаОбъект нужно добавить типы значений доступа, для которых предусмотрены типы групп значений доступа.

Внимание. Для того чтобы изменения вступили в силу сразу при разработке и отладке конфигурации, требуется обновить вспомогательные данные. См. раздел «[Обновление вспомогательных данных во время разработки](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_обновление_вспомогательных_данных)».

**Обязательные виды доступа**

К обязательным видам доступа, которые входят в состав подсистемы, относятся: Пользователи, ВнешниеПользователи.

В состав разрешенных пользователей вида доступа Пользователи автоматически включается текущий пользователь: аналогично в состав разрешенных внешних пользователей вида доступа ВнешниеПользователи автоматически включается текущий внешний пользователь.

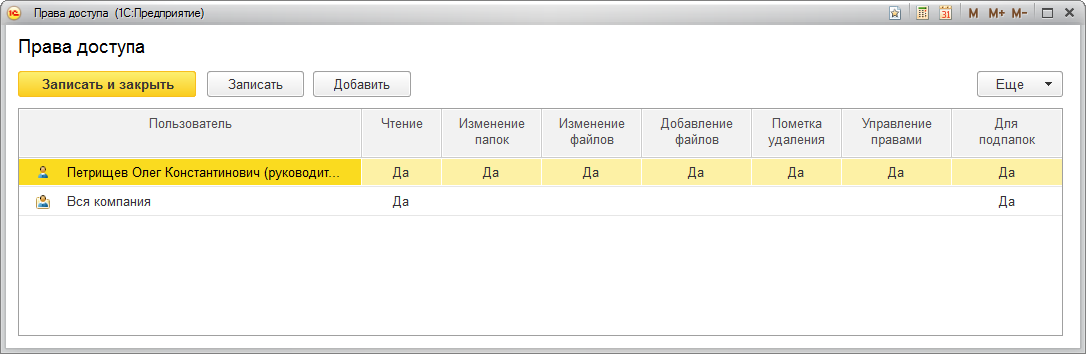
В [стандартных шаблонах ограничения доступа](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_разработка_ограничений_прав) (см. ниже) в качестве видов доступа также используются специальные функции ПравоЧтения, ПравоИзменения, Объект. Первые две функции доступны в шаблонах [ПоЗначениям](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_шаблон_«#позначениям»), [ПоЗначениямРасширенный](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_шаблон_«#позначениямрасширенный»), [ПоЗначениямИНаборамРасширенный](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_шаблон_«#позначенияминаборамрасшире). Функция Объект только в шаблоне [ПоЗначениямИНаборамРасширенный](https://its.1c.ru/db/content/bsp301doc/src/3.58.%20%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5%20%D0%B4%D0%BE%D1%81%D1%82%D1%83%D0%BF%D0%BE%D0%BC.htm?_=1542292879#_шаблон_«#позначенияминаборамрасшире). Эти функции тоже называются видами доступа, хотя их нет нигде, кроме стандартных шаблонов. Использование этих видов доступа рассмотрено ниже.

**Настройка функциональной опции «ОграничиватьДоступНаУровнеЗаписей<ИмяПрикладногоРешения>»**

Для реквизитов, созданных выше, необходимо добавить функциональную опцию ОграничиватьДоступНаУровнеЗаписей<ИмяПрикладногоРешения>, например ОграничиватьДоступНаУровнеЗаписейУправлениеТоровлей. В свойство Хранение задать константу ОграничиватьДоступНаУровнеЗаписей.

**Разработка прав для настройки прав отдельных объектов**

В случае необходимости настройки прав для отдельных объектов информационной базы, следует определить список типов этих объектов и в модуле УправлениеДоступомПереопределяемый и для каждого типа вписать права в процедуру ПриЗаполненииВозможныхПравДляНастройкиПравОбъектов, которые используются для автоматического создания формы настройки этих прав.



Например, для получения состава прав (колонок), как на рисунке, требуется следующий код:

Процедура ПриЗаполненииВозможныхПравДляНастройкиПравОбъектов(ВозможныеПрава) Экспорт

////////////////////////////////////////////////////////////

// Справочник.ПапкиФайлов

// Право "Чтение папок и файлов".

Право = ВозможныеПрава.Добавить();

Право.ВладелецПрав  = Метаданные.Справочники.ПапкиФайлов.ПолноеИмя();

Право.Имя           = "Чтение";

Право.Заголовок     = НСтр("ru = 'Чтение'");

Право.Подсказка     = НСтр("ru = 'Чтение папок и файлов'");

Право.НачальноеЗначение = Истина;

// Права для стандартных шаблонов ограничений доступа.

Право.ЧтениеВТаблицах.Добавить("\*");

// Право "Изменение папок"

Право = ВозможныеПрава.Добавить();

Право.ВладелецПрав  = Метаданные.Справочники.ПапкиФайлов.ПолноеИмя();

Право.Имя           = "ИзменениеПапок";

Право.Заголовок     = НСтр("ru = 'Изменение

                                 |папок'");

Право.Подсказка     = НСтр("ru = 'Добавление, изменение и

                                 |пометка удаления папок файлов'");

// Права, требуемые для этого права.

Право.ТребуемыеПрава.Добавить("Чтение");

// Права для стандартных шаблонов ограничений доступа.

Право.ИзменениеВТаблицах.Добавить(Метаданные.Справочники.ПапкиФайлов.ПолноеИмя());

// Право "Изменение файлов"

Право = ВозможныеПрава.Добавить();

Право.ВладелецПрав  = Метаданные.Справочники.ПапкиФайлов.ПолноеИмя();

Право.Имя           = "ИзменениеФайлов";

Право.Заголовок     = НСтр("ru = 'Изменение

                                 |файлов'");

Право.Подсказка     = НСтр("ru = 'Изменение файлов в папке'");

// Права, требуемые для этого права.

Право.ТребуемыеПрава.Добавить("Чтение");

// Права для стандартных шаблонов ограничений доступа.

Право.ИзменениеВТаблицах.Добавить("\*");

// Право "Добавление файлов"

Право = ВозможныеПрава.Добавить();

Право.ВладелецПрав  = Метаданные.Справочники.ПапкиФайлов.ПолноеИмя();

Право.Имя           = "ДобавлениеФайлов";

Право.Заголовок     = НСтр("ru = 'Добавление

                                 |файлов'");

Право.Подсказка     = НСтр("ru = 'Добавление файлов в папку'");

// Права, требуемые для этого права.

Право.ТребуемыеПрава.Добавить("ИзменениеФайлов");

// Право "Пометка удаления файлов".

Право = ВозможныеПрава.Добавить();

Право.ВладелецПрав  = Метаданные.Справочники.ПапкиФайлов.ПолноеИмя();

Право.Имя           = "ПометкаУдаленияФайлов";

Право.Заголовок     = НСтр("ru = 'Пометка

                                 |удаления'");

Право.Подсказка     = НСтр("ru = 'Пометка удаления файлов в папке'");

// Права, требуемые для этого права.

Право.ТребуемыеПрава.Добавить("ИзменениеФайлов");

Право = ВозможныеПрава.Добавить();

Право.ВладелецПрав  = Метаданные.Справочники.ПапкиФайлов.ПолноеИмя();

Право.Имя           = "УправлениеПравами";

Право.Заголовок     = НСтр("ru = 'Управление

                                 |правами'");

Право.Подсказка     = НСтр("ru = 'Управление правами папки'");

// Права, требуемые для этого права.

Право.ТребуемыеПрава.Добавить("Чтение");

КонецПроцедуры

* ● Указать состав типов как состав владельцев прав в процедуре ПриЗаполненииВозможныхПравДляНастройкиПравОбъектов:
* ● в определяемом типе ВладелецНастроекПрав,
* ● в определяемом типе ВладелецНастроекПравОбъект,
* ● в общей команде НастроитьПрава.
* ● Использовать в модуле УправлениеДоступом функцию ЕстьПраво для получения пользователем разрешения на права.
* ● В ограничениях доступа к данным, которые реализуются с использованием стандартных шаблонов ОДД (см. ниже), следует использовать специальный вид доступа НастройкиПрав.

Внимание. Для того чтобы изменения вступили в силу сразу при разработке и отладке конфигурации, требуется обновить вспомогательные данные. См. раздел «[Обновление вспомогательных данных во время разработки](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_обновление_вспомогательных_данных)».

**Настройка вычисления использования видов доступа в зависимости от настроек программы**

В процедуре ПриЗаполненииИспользованияВидаДоступа общего модуля УправлениеДоступомПереопределяемый можно установить код вычисления использования видов доступа. Например, код процедуры может выглядеть следующим образом:

// Заполняет использование видов доступа в зависимости от функциональных опций конфигурации,

// например "ИспользоватьГруппыДоступаНоменклатуры".

//

// Параметры:

//  ВидДоступа    - Строка - имя вида доступа заданное в процедуре ПриЗаполненииВидовДоступа.

//  Использование - Булево - начальное значение Истина.

//

Процедура ПриЗаполненииИспользованияВидаДоступа(ИмяВидаДоступа, Использование) Экспорт

Если ИмяВидаДоступа = "\_ДемоГруппыНоменклатуры" Тогда

Использование = Константы.\_ДемоОграничиватьДоступПоНоменклатуре.Получить();

ИначеЕсли ИмяВидаДоступа = "\_ДемоГруппыПартнеров" Тогда

Использование = Константы.\_ДемоОграничиватьДоступПоПартнерам.Получить();

ИначеЕсли ИмяВидаДоступа = "\_ДемоФизическиеЛица" Тогда

Использование = Константы.\_ДемоОграничиватьДоступПоФизическимЛицам.Получить();

КонецЕсли;

КонецПроцедуры

При изменении использования вида доступа требуется вызвать обновление разрешенных значений – например, из модуля менеджера значения константы:

Процедура ПриЗаписи(Отказ)

Если ОбменДанными.Загрузка Тогда

Возврат;

КонецЕсли;

УправлениеДоступом.ОбновитьРазрешенныеЗначенияПриИзмененииИспользованияВидовДоступа();

КонецПроцедуры

Для изменения действующего значения в ограничениях доступа к объектам требуется перезапуск сеанса.

**Разработка ролей и профилей групп доступа**

Для каждой спроектированной роли:

* ● добавить в конфигурацию объект метаданных «роль», установить имя и синоним; снять флажок Устанавливать права для новых объектов; снять флажок Независимые права подчиненных объектов;
* ● установить флажки напротив требуемых прав по каждому объекту метаданных конфигурации;
* ● у объектов метаданных, на которые установлены права, снять лишние флажки напротив требуемых прав по полям (реквизитам, табличным частям, измерениям, ресурсам м т. д.);
* ● для каждого права каждого объекта метаданных задать тексты ограничений доступа с использованием стандартных шаблонов. Применяемые в роли стандартные шаблоны следует предварительно скопировать из поставляемой роли ИзменениеУчастниковГруппДоступа.

При необходимости переименовать роль следует учитывать методику раздела «[Использование идентификаторов объектов метаданных](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_использование_идентификаторов_объек)».

Затем в режиме 1С:Предприятие необходимо создать требуемое количество профилей групп доступа пользователей, заполнив их ролями и видами доступа, созданными на предыдущем шаге.

Важно!

В профиль следует включать только те виды доступа, которые задействованы в ограничениях прав на списки, настроенные в ролях, выбранных для профиля. Настройка видов доступа в профиле и группе доступа не влияет на ограничение прав на те списки, в ограничении которых виды доступа не задействованы.

В профиле возможно использовать предустановленные виды доступа. Вариант «предустановленный» обозначает, что вид доступа не настраивается в группе доступа, а настраивается только в профиле для всех групп доступа. Среди видов доступа могут быть те, которые имеют смысл больше как предустановленные, нежели настраиваемые в группе доступа. Предустанавливать в профиле можно любой вид доступа, если это удобно (или требуется) с точки зрения администрирования групп доступа этого профиля. Так, виды доступа, которые предназначены, например, для «разрезания» одной таблицы по признаку ХозяйственнаяОперация, логично использовать предустановленными. Для кассира – свой список хозопераций, для подотчетного лица – свой, а для бухгалтера – всех хозопераций по отношению, например, к документам Расходный кассовый ордер.

Затем надо вписать в процедуру ПриЗаполненииПоставляемыхПрофилейГруппДоступа общего модуля УправлениеДоступомПереопределяемый код по созданию и обновлению поставляемых профилей групп доступа при начальном заполнении и обновлении информационной базы. Для автоматического формирования текста этой процедуры следует применять обработку УправлениеДоступом.epf из демонстрационной конфигурации, которая формирует код на основании профилей групп доступа, существующих в информационной базе.

**Разработка ограничений прав доступа**

Для установки текстов ограничений прав доступа используется один из четырех стандартных шаблонов:

#ПоЗначениям(…)

#ПоНаборамЗначений(…)

#ПоЗначениямРасширенный(…)

#ПоЗначениямИНаборамРасширенный(…)

Примечание

Подробное описание назначения и параметров шаблонов приведено в комментариях в начале текстов самих шаблонов. Рекомендуется изучить это описание перед прочтением дальнейшего материала. Стандартные шаблоны поставляются в роли ИзменениеУчастниковГруппДоступа.

Наличие сразу нескольких шаблонов объясняется необходимостью решать различные задачи. Так, шаблоны #ПоЗначениям и #ПоНаборамЗначений позволяют решить большую часть простых задач.

Для шаблонов #ПоЗначениямИНаборамРасширенный (только для вида доступа Объект) и #ПоНаборамЗначений можно использовать модификатор РасширенноеИЛИ, который меняет расчет ограничений с несколькими проверками, результаты которых объединяются по ИЛИ. Например, электронное письмо доступно, если разрешена Учетная запись или Организация. Без модификатора, если ограничение по учетным записям или организациям не используется, все письма будут разрешены. Случаи неиспользования: выключена константа использования вида доступа, нет вида доступа в профиле.

Шаблоны с модификатором РасширенноеИЛИ работают медленнее, поэтому не следует использовать модификатор там, где нет логики ограничения по ИЛИ. Если в ограничении используется вид доступа Объект, а типы объектов и их логика ограничения заранее неизвестны, модификатор нужно указать, когда требуется расширенный расчет ИЛИ.

При установке ограничений прав доступа в ролях, следует указывать их для всех полей (в колонке Поля выбирать вариант <Прочие поля> без указания ограничений для конкретных полей). Это обусловлено тем, что для пользователя состав строк в списках и отчетах не должен зависеть от состава колонок, которые он настроил для отображения тем или иным спсобом. В частности, безусловно запрещенные поля (например, для всех кроме администратора), следует вынести в отдельный список, на который назначить нужные права. Безусловно разрешенные поля делать не требуется (ранее для работы механизмов 1С:Предприятия требовалось предоставить доступ к некоторым стандартным реквизитам, например, Код, этого более не требуется, так как теперь 1С:Предприятие использует для технологического доступа привилегированный режим).

**Шаблон «#ПоЗначениям»**

Шаблон #ПоЗначениям позволяет реализовать проверку разрешения значений доступа в полях таблицы (указывать виды доступа нужно, чтобы текст ограничения автоматически оптимизировался, когда вид доступа не используется). Например, текст ограничения доступа вида:

#ПоЗначениям("Документ.ОприходованиеТоваров", "", "",

"Организации", "Организация",

"Склады", "Склад", …).

ограничивает чтение таким образом, что пользователь увидит в списке документов Оприходование товаров только те документы, у которых в реквизите Организация есть значение, разрешенное пользователю по виду доступа Организации, и в реквизите Склад есть значение, разрешенное по виду доступа Склады. Причем значения должны быть разрешены в группе доступа пользователей одновременно, а профиль групп доступа пользователей должен включать роль с правом чтения документов Оприходование товаров.

В шаблоне можно выполнить максимум 16 проверок.

Этот и другие шаблоны поддерживают передачу условия на языке запросов. Для этого вместо имени вида доступа нужно указать имя Условие, а вместо имени поля – текст на языке запросов. Например:

#ПоЗначениям("Документ.ОприходованиеТоваров", "", "",

"Организации", "Организация",

"Склады", "Склад",

"Условие", "Т.ХозяйственнаяОперация = Значение(Перечисление.ХозяйвственныеОперации.Инветаризация) ", …),

В этом примере добавляется условие: документ доступен только при хозяйственной операции Инвентаризация. Из этого следует, что при другой хозяйственной операции, например Прочее, документ не будет доступен ни одному пользователю, кроме пользователей с ролью Полные права.

**Шаблон «#ПоНаборамЗначений»**

В первую очередь шаблон #ПоНаборамЗначений предназначен для ограничения чтения журналов документов, у которых документы имеют различные ограничения чтения, например по составу реквизитов. Обычно для журнала приходится делать анализ типа документа, а затем применять те же проверки, что и для документа соответствующего типа, т. е. повторять логику ограничений всех документов в каждом журнале, в который входят документы. Как правило, этот способ очень «дорогой» в плане производительности и трудоемок с точки зрения разработки и сопровождения. Поэтому целесообразно предварительно записывать значения доступа объектных элементов данных, необходимые для работы логики ограничения доступа, в специальный регистр – НаборыЗначенийДоступа. Такие данные называются наборами значений доступа.

Например, если для документа Оприходование товаров в регистре НаборыЗначенийДоступа имеются записи:

<Документ1>, , "Организации", "Организация1"

<Документ1>, , "Склады", "Склад1";

то для их использования текст ограничения права чтения журнала, содержащего этот документ, должен быть таким:

#ПоНаборамЗначений("ЖурналДокументов.СкладскиеДокументы", "", "", "Ссылка"),

где Ссылка – имя стандартного реквизита журнала.

В этом случае логика работы шаблона #ПоНаборамЗначений такова: по переданному значению стандартного реквизита Ссылка строки журнала в регистре Наборы значений доступа будут найдены все строки, содержащие Значение доступа, и выполнена проверка, что все значения доступа одновременно разрешены пользователю в одной из его групп доступа.

Помимо журналов документов шаблон #ПоНаборамЗначений может применяться для реализации логики ограничения доступа и к другим объектам метаданных. При этом использование заранее подготовленных наборов значений позволяет избежать тех ограничений, которые накладываются языком запросом. Применение шаблона #ПоНаборамЗначений требует дополнительной настройки объекта метаданных. К объекту метаданных необходимо добавить табличную часть Наборы значений доступа с реквизитами, совпадающими с измерениями и ресурсами регистра сведений Наборы значений доступа. В тексте ограничения доступа в качестве четвертого параметра шаблона #ПоНаборамЗначений указывается пустая строка. Также требуется указать типы этих объектов в подписке на событие ЗаполнитьНаборыЗначенийДоступаТабличныхЧастей (для справочников) через определяемый тип ВладелецСОграничениемПоНаборамЗначенийДоступаОбъект или в дополнительно созданной подписке на события, имя которой начинается, как имя поставляемой подписки, а обработчик совпадает (только для документов).

**Шаблон «#ПоЗначениямРасширенный»**

В некоторых случаях возможностей шаблонов #ПоЗначеним и #ПоНаборамЗначений бывает недостаточно. Например, требуется ограничить документ Перемещение товаров так, чтобы чтение документа было доступно пользователю исходя из разрешенности Организации и СкладаПолучателя или СкладаОтправителя, а запись документа была доступна только при разрешенности всех трех: Организации, СкладаПолучателя и СкладаОтправителя. В этом случае право Запись можно ограничить, применив шаблон #ПоЗначениям, а право Чтение – нет.

Для ограничений, предполагающих логику с ИЛИ, предусмотрены расширенные шаблоны, в частности #ПоЗначениямРасширенный. Например, ограничение для права Чтение выглядит следующим образом:

#ПоЗначениямРасширенный("Документ.ПеремещениеТоваров", "", "",

"",

"",

"Организации", "Т.Организация", "И(",

"Склады", "Т.СкладОтправитель", "ИЛИ",

"Склады", "Т.СкладПолучатель", ")", …)

Требуется явно указывать псевдоним текущей таблицы «Т.» перед полем.

Кроме реализации логики ИЛИ расширенные шаблоны могут использоваться для проверки реквизитов табличных частей – через присоединяемые таблицы. Для того чтобы к основной таблице присоединить табличную часть, в четвертый параметр шаблона нужно передать текст на языке запросов, например:

"Внутреннее Соединение Документ.ПеремещениеТоваров.Номенклатура

КАК Т2 по Т.Ссылка = Т2.Ссылка",

Для шаблона основная таблица расширяется полями табличной части, а за счет явного применения псевдонимов их можно указывать, подобно полям основной таблицы. Т. е. в шаблон можно добавить проверку реквизита табличной части, например:

"ГруппыНоменклатуры", "Т2.Номенклатура", ""

В результате получится такое ограничение:

#ПоЗначениямРасширенный("Документ.ПеремещениеТоваров", "", "",

"Внутреннее Соединение Документ.ПеремещениеТоваров.Номенклатура КАК Т2 по Т.Ссылка = Т2.Ссылка",

"",

"Организации", "Т.Организация", "И(",

"Склады", "Т.СкладОтправитель", "ИЛИ",

"Склады", "Т.СкладПолучатель", ")И",

"ГруппыНоменклатуры", "Т2.Номенклатура", "", …)

Когда основная таблица расширяется полями присоединенной таблицы, происходит умножение строк. Например, табличная часть содержит 2 строки c разной номенклатурой. В этом случае документ будет доступен, если выполнились условия проверки основной таблицы («шапки» документа) и доступна хотя бы одна из двух номенклатур табличной части.

**Шаблон «#ПоЗначениямИНаборамРасширенный»**

Для решения задачи ограничения доступа к объектам метаданных в зависимости от ограничений к другим объекта может использоваться шаблон #ПоЗначениямИНаборамРасширенный.

Например, файл (элемент справочника Файлы), прикрепленный к документу Поступление товаров и услуг, должен быть доступен пользователю при тех же условиях, что и сам документ.

Такая задача может быть решена «повтором» логики ограничения документа Поступление товаров и услуг в ограничении справочника Файлы. Однако на практике это приводит к необходимости синхронно изменять логику ограничения документов и файлов. Кроме того, поскольку элементы справочника Файлы могут «прикрепляться» к произвольному числу объектов метаданных (например, к папкам файлов, документам Заказ, Расходная накладная и т. п.), то задача реализации логики ограничения доступа к файлам сравнима по сложности с ограничением доступа к журналу документов, который содержит все виды документов конфигурации.

В некоторых случаях, когда число владельцев небольшое (не более 10), можно повторять логику ограничения владельца и при этом иметь приемлемые показатели производительности, особенно, если логика ограничения владельцев совпадает. При повторении логики ограничения нужно учесть, что проверить значения доступа владельцев недостаточно, а нужно еще учесть наличие прав на таблицу владельца, иначе будут доступны файлы владельцев, которые сами недоступны по правам (по составу ролей). Для этой цели нужно использовать специальные виды доступа ПравоЧтения и ПравоИзменения, которые позволяют проверить соответственно наличие прав Чтение и Изменение на таблицу владельца, например:

#ПоЗначениям("Справочник.Файлы", "Чтение", "",

"ПравоЧтение", "Т.ВладелецФайла",

"Организации", "Т.ВладелецФайла.Организация", …).

Когда владельцев много, этот способ не подходит. Для такого случая, подсистема «Управление доступом» предоставляет специальный вид доступа по объекту-владельцу. Для использования этого вида доступа в шаблоне следует указывать в качестве имени вида доступа Объект. Вид доступа Объект предполагает, что в поле находится значение, которое является ссылкой на объект данных (например, документ). А в регистре Наборы значений доступа, как и для шаблона #ПоНаборамЗначений, записаны наборы значений для этого объекта, которые следует проверить. Исключение составляют владельцы настроек прав – для них наборы значений доступа записывать не требуется, но процедура заполнения может понадобиться, если они используются для заполнения наборов значений доступа зависимых объектов.

Так, используя шаблон #ПоЗначениямИНаборамРасширенный, можно проверить не одно значение доступа, а набор значений, подготовленный при записи документа, ссылка на который находится в реквизите, например, Владелец элемента справочника Файлы.

#ПоЗначениямИНаборамРасширенный("Справочник.Файлы", "", "",

"",

"",

"Объект", "Т.Владелец" , "", …)

Наборы значений элементов справочника Файлы требуют связанного обновления при изменении, например, наборов значений доступа документа – владельца файлов Поступление товаров и услуг.

При использовании вида доступа Объект кроме проверки наборов выполняется еще проверка прав на объект (то есть добавляется проверка, которая используется в видах доступа ПравоЧтения и ПравоИзменения). В ограничении права чтения таблицы выполняется проверка права чтения объекта и наборов ограничения чтения объекта, а в ограничении прав добавления, изменения, удаления таблицы выполняется проверка права изменения объекта и наборов ограничения изменения объекта.

Права объекта, которые проверяются в ограничении по виду доступа Объект, можно переопределить в процедуре ПриЗаполненииЗависимостейПравДоступа модуля УправлениеДоступомПереопределяемый.

Структура регистра Наборы значений доступа поддерживает логику ИЛИ для значений доступа и различные наборы по правам доступа. Для реализации логики ИЛИ добавлено измерение Номер набора типа Число, а для наборов по правам добавлены два ресурса: Чтение, Изменение – типа Булево.

Сложную логику ИЛИ записать в регистр «напрямую» не получится. Номер набора позволяет сделать для одного ссылочного элемента данных (например, документа) несколько наборов значений, по которым проверяется доступность документа. Причем достаточно, чтобы значения хотя бы одного набора значений были доступны одновременно. Точнее, результаты проверки значений доступа по видам доступа в пределах набора с одним номером объединяются по И, а результаты проверки целых наборов объединяются по ИЛИ. Например, для документа Перемещение товаров регистр Наборы значений доступа будет содержать записи для логики права Чтение:

<Документ2>, 0, "Организации", "Организация1"

<Документ2>, 0, "Склады", "Склад1";

<Документ2>, 1, "Организации", "Организация1"

<Документ2>, 1, "Склады", "Склад2";

Видно, что, хотя значений доступа три: Организация1, Склад1, Склад2, – записей в регистре четыре. Это связано с раскрытием скобок в логическом выражении О1 и (С1 или С2), что приводит к логическому выражению суммы произведений О1 и С1 или О1 и С2.

Действие раскрытия скобок первичного логического выражения, объединяющего результаты проверки значений доступа, необходимо для заполнения наборов записей регистра сведений Наборы значений доступа.

Записи для логики права Изменение:

<Документ2>, 0, "Организации", "Организация1"

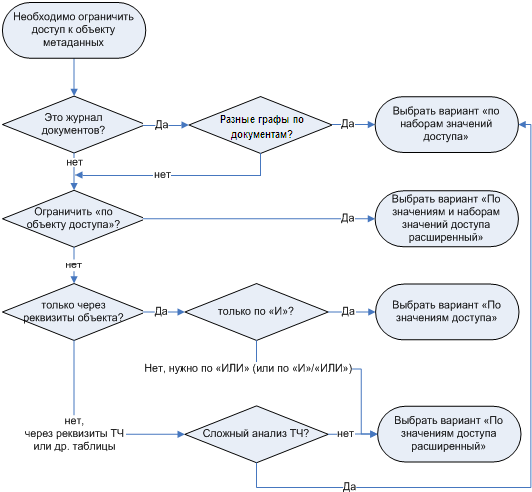
<Документ2>, 0, "Склады", "Склад1";

<Документ2>, 0, "Склады", "Склад2";

Пример процедуры заполнения приведен ниже.

Шаблон #ПоЗначениямИНаборамРасширенный отличается от шаблона #ПоЗначениямРасширенный только поддержкой специального вида доступа Объект.

Для быстрого выбора шаблона можно использовать схему-алгоритм.



**Разработка процедур «Заполнить наборы значений доступа»**

При использовании стандартных шаблонов #ПоНаборамЗначений или #ПоЗначениямИНаборамЗначений с видом доступа Объект потребуется записать в регистр Наборы значений доступа сведения, требуемые для их работы.

Запись наборов значений доступа выполняется при записи объектов – например, документов. Перед записью вызывается пользовательская процедура заполнения наборов значений доступа.

Процедура заполнения помещается в модуль объекта, и его тип включается в подписку на события Записать наборы значений доступа и в состав типов измерения Объект регистра сведений Наборы значений доступа.

При заполнении можно использовать только следующие свойства таблицы:

* ● Номер набора (если набор один, то заполнять не нужно);
* ● Вид доступа (только для специальных видов доступа ПравоЧтения, ПравоИзменения);
* ● Значение доступа (назначить явно);
* ● Чтение, Изменение (если набор для всех прав, то заполнять не нужно).

Количество заполненных записей должно быть больше нуля, иначе будет вызвано исключение. Это важно, т. к. процедура заполнения наборов значений доступа после включения режима Ограничивать доступ на уровне записей вызывается из регламентного задания Заполнение данных для ограничения доступа для тех объектов, которые имеют пустой набор записей в регистре Наборы значений доступа. При выключении режима Ограничивать доступ на уровне записей производится запись пустого набора. Следовательно, при включении режима выполнится заполнение очищенных наборов (или полное заполнение). Регламентное задание вызывается многократно и отрабатывает порцию заполнения. Очищение наборов происходит «по ходу» перезаписи объектов.

Технология порционного заполнения/очистки использует пустой набор записей как флажок состояния. Когда нужно записать заведомо «запрещенный набор», записывается не пустой набор, а следующий:

Строка = Таблица.Добавить();

Строка.ЗначениеДоступа = Перечисления.ДополнительныеЗначенияДоступа.ДоступЗапрещен;

Когда нужно записать заведомо «разрешенный набор», записывается не пустой набор, а следующий:

Строка = Таблица.Добавить();

Строка.ЗначениеДоступа = Перечисления.ДополнительныеЗначенияДоступа.ДоступРазрешен;

Пример процедуры:

* ● ЗаполнитьНаборыЗначенийДоступа в модуле объекта (тип объекта включается в подписку на событие ЗаписатьНаборыЗначенийДоступа (через определяемый тип ВладелецНаборовЗначенийДоступаОбъект) или в дополнительно созданную подписку на события, имя которой начинается, как имя поставляемой подписки, а обработчик совпадает).

Процедура ЗаполнитьНаборыЗначенийДоступа(Таблица) Экспорт

Строка = Таблица.Добавить();

Строка.Изменение       = Истина;

Строка.ЗначениеДоступа = Организация1;

Строка = Таблица.Добавить();

Строка.ЗначениеДоступа = Склад1;

Строка = Таблица.Добавить();

Строка.ЗначениеДоступа = Склад2;

Строка = Таблица.Добавить();

Строка.НомерНабора     = 1;

Строка.Чтение          = Истина;

Строка.ЗначениеДоступа = Организация1;

Строка = Таблица.Добавить();

Строка.НомерНабора     = 1;

Строка.ЗначениеДоступа = Склад1;

Строка = Таблица.Добавить();

Строка.НомерНабора     = 2;

Строка.Чтение          = Истина;

Строка.ЗначениеДоступа = Организация1;

Строка = Таблица.Добавить();

Строка.НомерНабора     = 2;

Строка.ЗначениеДоступа = Склад2;

КонецПроцедуры

Если в процедуре заполнения не задан номер набора, то считается, что добавлен один набор с номером 1. Если в пределах набора с одним номером не указано ни одно из прав Чтение или Изменение, то считается, что набор относится ко всем правам.

В наборе допускается только одна запись одним и тем же значением доступа, дубли строк и наборов удаляются автоматически в процедуре ДобавитьНаборыЗначенийДоступа, которая вызывается для этих целей перед записью наборов значений доступа в регистр сведений НаборыЗначенийДоступа со сжатием наборов (лишние наборы удаляются по правилам упрощения логических функций).

Внимание. Для того чтобы изменения вступили в силу сразу при разработке и отладке конфигурации, требуется обновить вспомогательные данные. См. раздел «[Обновление вспомогательных данных во время разработки](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_обновление_вспомогательных_данных)».

**Зависимые наборы значений доступа**

В случае, когда наборы значений доступа нужно сформировать с учетом доступа к другому объекту (к примеру, доступ к бизнес-процессу \_ДемоЗаданиеСРолевойАдресацией определяется по доступу к его объекту-основанию, например по элементу справочника Файлы), появляется необходимость включать в состав одних наборов другие наборы.

Для реализации включения одних наборов в другие разработана процедура ДобавитьНаборыЗначенийДоступа общего модуля УправлениеДоступом, которая добавляет в таблицу-приемник Приемник наборы из таблицы-источника Источник либо логическим сложением, либо логическим умножением. Например, наборы задачи:

Процедура ЗаполнитьНаборыЗначенийДоступа(Таблица) Экспорт

Строка = Таблица.Добавить();

Строка.ЗначениеДоступа = Исполнитель;

// Создание пустой таблицы.

ТаблицаБизнесПроцесса = УправлениеДоступом.ТаблицаНаборыЗначенийДоступа();

// Заполнение наборов бизнес-процесса.

УправлениеДоступом.ЗаполнитьНаборыЗначенийДоступа(БизнесПроцесс, ТаблицаБизнесПроцесса);

// Логическое сложение наборов в таблицах (по «ИЛИ»).

УправлениеДоступом.ДобавитьНаборыЗначенийДоступа(Таблица, ТаблицаБизнесПроцесса);

КонецПроцедуры

Следует разработать процедуры ЗаполнитьНаборыЗначенийДоступа для всех объектов, которые могут быть объектом-основанием или задействованы при формировании наборов значений доступа.

Могут быть объекты, у которых наборы значений доступа не записываются в регистр сведений НаборыЗначенийДоступа, а используются только при формировании других записываемых наборов значений доступа. Типы таких объектов нужно включить в подписку на события ЗаписатьЗависимыеНаборыЗначенийДоступа (через определяемый тип ВладелецВнешнихЗначенийВНаборахЗначенийДоступаОбъект) или в дополнительно созданную подписку на события, имя которой начинается, как имя поставляемой подписки, а обработчик совпадает.

Для реализации зависимостей необходимо разработать логику вызова перезаписи наборов значений доступа зависимых объектов в обработчике ПриИзмененииНаборовЗначенийДоступа общего модуля УправлениеДоступомПереопределяемый, например:

Процедура ПриИзмененииНаборовЗначенийДоступа(Знач Ссылка, СсылкиНаЗависимыеОбъекты) Экспорт

ПолноеИмя = Ссылка.Метаданные().ПолноеИмя();

Если ПолноеИмя = "Задача.ЗадачаИсполнителя" Тогда

// Зависимые типы объектов:

//  БизнесПроцесс.Задание

Запрос = Новый Запрос(

"ВЫБРАТЬ

| Задание.Ссылка

|ИЗ

| БизнесПроцесс.Задание КАК Задание

|ГДЕ

| Задание.Предмет = &Предмет");

Запрос.УстановитьПараметр("Предмет", Ссылка);

СсылкиНаЗависимыеОбъекты = Запрос.Выполнить().Выгрузить().ВыгрузитьКолонку("Ссылка");

ИначеЕсли ПолноеИмя = "БизнесПроцесс.Задание" Тогда

// Зависимые типы объектов:

//  БизнесПроцесс.Задание

// через Задача.ЗадачаИсполнителя в основании.

Запрос = Новый Запрос(

"ВЫБРАТЬ РАЗЛИЧНЫЕ

| Задание.Ссылка

|ИЗ

| БизнесПроцесс.Задание КАК Задание

| ВНУТРЕННЕЕ СОЕДИНЕНИЕ Задача.ЗадачаИсполнителя КАК ЗадачаИсполнителя

| ПО Задание.Предмет = ЗадачаИсполнителя.Ссылка

| И (ЗадачаИсполнителя.БизнесПроцесс = &БизнесПроцесс)");

Запрос.УстановитьПараметр("БизнесПроцесс", Ссылка);

СсылкиНаЗависимыеОбъекты = Запрос.Выполнить().Выгрузить().ВыгрузитьКолонку("Ссылка");

КонецЕсли;

КонецПроцедуры

Специальные имена видов доступа ПравоЧтения, ПравоИзменения требуются, когда нужно разработать ограничение с зависимостью по правам «подчиненного» объекта от «ведущего» объекта. В шаблонах ОДД в проверку по виду доступа Объект встроена логика проверки зависимых прав, как и в этих видах доступа. В связи с этим сочетать данные виды доступа с видом доступа Объект не требуется. Также не требуется использовать эти виды доступа при заполнении зависимых наборов значений доступа. Для этих целей предусмотрен третий параметр в процедуре ЗаполнитьНаборыЗначенийДоступа, при указании которого соответствующие проверки добавляются автоматически.

Внимание. Для того чтобы изменения вступили в силу сразу при разработке и отладке конфигурации, требуется обновить вспомогательные данные. См. раздел «[Обновление вспомогательных данных во время разработки](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_обновление_вспомогательных_данных)».

**Разработка процедур ПриЗаполненииОграниченияДоступа**

При разработке и внесении изменений в логику ограничений с использованием стандартных шаблонов следует синхронно вносить изменения в процедуру ПриЗаполненииОграниченияДоступа модулей менеджеров и запускать отчет ПроверкаВнедренияБСП.erf (с флажком Исправлять ошибки и с отбором по подсистеме Управление доступом) для обновления ограничений в ролях, а также определяемых типов и предопределенных элементов в справочнике ИдентификаторыОбъектовМетаданных.

Описания логики ограничения в процедуре ПриЗаполненииОграниченияДоступа на простых примерах.

● Организация и контрагент в шапке документа:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ЗначениеРазрешено(Организация)

|И ЗначениеРазрешено(Контрагент)";

● Организация в шапке документа, контрагент в табличной части, достаточно одного разрешенного контрагента:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ЗначениеРазрешено(Организация)

|И ЗначениеРазрешено(Поставщики.Контрагент)";

● Организация в шапке документа, контрагент в табличной части, достаточно одного разрешенного контрагента (если табличная часть пустая, тогда доступ по контрагенту разрешен):

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ЗначениеРазрешено(Организация)

|И ЗначениеРазрешено(Поставщики.Контрагент, Null КАК Истина)";

● Организация в шапке документа, контрагент в табличной части и требуется, чтобы все контрагенты были разрешены (если табличная часть пустая, тогда доступ по контрагенту запрещен):

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ЗначениеРазрешено(Организация)

|И ДляВсехСтрок(ЗначениеРазрешено(Поставщики.Контрагент))";

● Организация и контрагент в табличной части, при этом достаточно, чтобы любая из пар организации с контрагентом была разрешена:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ЗначениеРазрешено(Поставщики.Организация)

|И ЗначениеРазрешено(Поставщики.Контрагент)";

● Организация и контрагент в табличной части, при этом требуется, чтобы все пары организации с контрагентом были разрешены:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ДляВсехСтрок(

|      ЗначениеРазрешено(Поставщики.Организация)

|    И ЗначениеРазрешено(Поставщики.Контрагент))";

● Организация и контрагент в табличной части, при этом требуется, чтобы одна из организаций и один из контрагентов были разрешены:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ДляОднойИзСтрок(ЗначениеРазрешено(Поставщики.Организация))

|И ДляОднойИзСтрок(ЗначениеРазрешено(Поставщики.Контрагент))";

● Отправитель – измерение составного типа, при этом требуется проверять только ссылки Справочник.Склады:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

|  ЗначениеРазрешено(Отправитель ТОЛЬКО Справочник.Склады)";

● Организация в шапке, головная организация в справочнике организаций, при этом требуется, чтобы чтение было разрешено, когда разрешена организация в шапке или разрешена организация, у которой эта организация является головной, а изменение было разрешено, когда разрешена организация в шапке:

Ограничение.Текст =

"ПрисоединитьДополнительныеТаблицы

|ЭтотСписок КАК ЭтотСписок

|

|ЛЕВОЕ СОЕДИНЕНИЕ Справочник.Организации КАК Владельцы

| ПО Владельцы.Ссылка = ЭтотСписок.Владелец

|

|ЛЕВОЕ СОЕДИНЕНИЕ Справочник.Организации КАК ОбособленныеПодразделения

| ПО ОбособленныеПодразделения.ГоловнаяОрганизация = Владельцы.Ссылка

|;

|РазрешитьЧтение

|ГДЕ

| ЗначениеРазрешено(Владелец)

| ИЛИ ЗначениеРазрешено(ОбособленныеПодразделения.Ссылка)

|;

|РазрешитьИзменениеЕслиРазрешеноЧтение

|ГДЕ

| ЗначениеРазрешено(Владелец)";

Синтаксис описания логики ограничения в процедуре ПриЗаполненииОграниченияДоступа:

Ограничение.Текст =

"ПрисоединитьДополнительныеТаблицы

|ЭтотСписок КАК <Псевдоним>

|ЛЕВОЕ СОЕДИНЕНИЕ <Любая таблица> КАК <Псевдоним>

|ПО <Простое условие>

|… <Другие соединения>

|;

|РазрешитьЧтение

|ГДЕ

|        <Условие>

|  И/ИЛИ НЕ <Условие>

|  И/ИЛИ ВЫБОР

|          КОГДА <Условие>

|            ТОГДА <Условие>

|            ИНАЧЕ <Условие>

|         КОНЕЦ

|;

|РазрешитьИзменениеЕслиРазрешеноЧтение

|ГДЕ

|   <Условие> …";

Ограничение.ТекстДляВнешнихПользователей =

"<аналогично тексту для пользователей>";

<Условие> может быть в виде специальных функций ограничения:

● ЗначениеРазрешено(<Реквизит> [<проверяемые типы>] [, <уточнение сравнения 1> [, <уточнение сравнения 2>] …]),

где проверяемые типы могут быть (значения непроверяемых типов запрещены, если не уточнены отдельно):

● ТОЛЬКО <Имя таблицы>

● ТОЛЬКО (<Имя таблицы 1>, <Имя таблицы 2>, … )

● КРОМЕ <Имя таблицы>

● КРОМЕ (<Имя таблицы 1>, <Имя таблицы 2>, … )

где уточнение сравнения может быть:

● ПустаяСсылка КАК Ложь/Истина

● Неопределено КАК Ложь/Истина

● Null КАК Ложь/Истина

● Отключено КАК Ложь/Истина (только для функции ЗначениеРазрешено)

● <Таблица> КАК Ложь/Истина (например, Справочник.Проекты КАК Истина).

● ЧтениеОбъектаРазрешено (<также, как для функции ЗначениеРазрешено>)

● ЧтениеСпискаРазрешено (<также, как для функции ЗначениеРазрешено>)

● ИзменениеОбъектаРазрешено (<также, как для функции ЗначениеРазрешено>)

● ИзменениеСпискаРазрешено (<также, как для функции ЗначениеРазрешено>),

● последние 2 функции можно указать только в разделе РазрешитьИзменениеЕслиРазрешеноЧтение,

● ЭтоАвторизованныйПользователь (<также, как для функции ЗначениеРазрешено>).

Пример, как выглядит использование расширенных свойств сравнения в специальных функциях:

● ЗначениеРазрешено (Владелец ТОЛЬКО Справочник.Организации)

● ЗначениеРазрешено (Владелец КРОМЕ (Справочник.Проекты, Справочник.Номенклатура), ПустаяСсылка КАК Истина)

● ЗначениеРазрешено (Владелец, Неопределено КАК Истина)

<Условие> предоставляет следующие стандартные виды сравнения:

● <Реквизит> Равно | Не равно <предопределенное значение>

● <Реквизит> В списке | Не в списке <предопределенное значение 1>, …

● <Реквизит> ЕСТЬ NULL

● ТипЗначения (<Реквизит>) = Тип (<Имя таблицы>)

Пример, как выглядит использование стандартных видов сравнения:

● ВидПоставщика = Значение(Перечисления.ВидыПоставщиков.Основной)

● ВидПоставщика В (Значение(Перечисления.ВидыПоставщиков.Основной),

● Значение(Перечисления.ВидыПоставщиков.Дополнительный))

<Условие> также представлено двумя модификаторами, которые указывают способ сложения результатов проверок (столбец результатов), выполненных в условии, если использовался реквизит табличной части или дополнительной таблицы, который может содержать несколько значений (столбец значений):

● ДляВсехСтрок(<Условие>) – объединить результаты проверок в строках с помощью логического «И».

● ДляОднойИзСтрок(<Условие>) – объединить результаты проверок в строках с помощью логического «ИЛИ» (это поведение по умолчанию, но есть случаи, когда поведения по умолчанию недостаточно).

Описание логики RLS решает проблему реализации расширенного ИЛИ. Для этой цели введено специальное значение Отключено, которое можно уточнить "КАК Ложь" в функции ЗначениеРазрешено, (по умолчанию Отключено значить разрешено, то есть Истина). Когда в реквизите обнаружено значение доступа того вида доступа, который отключен по функциональной опции или не указан в профиле группы доступа, это значение доступа считается значением Отключено. При использовании уточнения "КАК Ложь" специального значения Отключено, следующее ограничение будет работать по оставшемуся виду доступа, когда один из них отключен, а не выдвать конечный результат разрешено:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ

| ЗначениеРазрешено(УчетнаяЗаписьЭлектроннойПочты, Отключено КАК Ложь)

| ИЛИ ЗначениеРазрешено(ОтветственныйПользователь, Отключено КАК Ложь)";

Синтаксис в зависимости от наличия ограничений по правам.

● Нет ограничения Чтения и нет ограничения Изменения:

Ограничение.Текст =

"";

● Одинаковые ограничения Чтения и Изменения:

Ограничение.Текст =

"РазрешитьЧтениеИзменение

|ГДЕ ... ";

● Разные ограничения Чтения и Изменения:

Ограничение.Текст =

"РазрешитьЧтение

|ГДЕ ...

|;

|РазрешитьИзменениеЕслиРазрешеноЧтение

|ГДЕ ... ";

● Нет ограничения Чтения, а есть только ограничение Изменения:

Ограничение.Текст =

"РазрешитьЧтение

|ГДЕ ИСТИНА

|;

|РазрешитьИзменениеЕслиРазрешеноЧтение

|ГДЕ ... ";

Ограничения в ролях реализуются на основе описания ограничения в модулях менеджеров с помощью одно из двух специальных шаблонов:

#ДляОбъекта("<Необязательно имя поля>")

#ДляРегистра("<Описание регистра>", "<Имя поля 1>", "<Необязательное имя поля 2>", "<…>", "<…>", "<…>")

В качестве первого параметра шаблона ДляРегистра используется, либо имя предопределенного элемента справочника ИдентификаторыОбъектовМетаданных, либо справочника ИдентификаторыОбъектовРасширений в формате "<Имя справочника>.<Полное имя регистра без точки>".

Для разработки ограничений доступа, которые указываются в модулях менеджеров объектов метаданных, следует использовать инструмент разработчика УправлениеДоступом.epf, который еще показывает и вариант шаблона ограничения для роли с параметрами, и другие сведения, необходимые для ручного обновления внедрения на основе указанного текста ограничения.

Для автоматической проверки (и обновления) вариантов шаблонов для ролей и их параметров, а также самих шаблонов в ролях, состава определяемых типов и предопределенных элементов справочника ИдентификаторыОбъектовМетаданных, следует использовать отчет ПроверкаВнедренияБСП.erf с отбором по подсистеме Управление доступом (в режиме исправления), который выполняет указанные действия в тех ролях, где в ограничениях прав доступа указаны стандартные шаблоны. При этом ограничения с помощью стандартных шаблонов не изменяются, а "оборачиваются" условием:

#Если &ОграничениеДоступаНаУровнеЗаписейУниверсально #Тогда

#ДляОбъекта(…)

#Иначе

#ПоЗначениям(… «Организации», «Организация», …)

#КонецЕсли

Перевод ключевых слов, используемых в ограничении доступа, которых нет в языке запросов 1С:Предприятия.

|  |  |
| --- | --- |
| Ключевое слово на русском языке | Ключевое слово на английском языке |
| * ● ПрисоединитьДополнительныеТаблицы * ● ЭтотСписок * ● РазрешитьЧтениеИзменение * ● РазрешитьЧтение * ● РазрешитьИзменениеЕслиРазрешеноЧтение * ● ЗначениеРазрешено * ● ЧтениеОбъектаРазрешено * ● ИзменениеОбъектаРазрешено * ● ЧтениеСпискаРазрешено * ● ИзменениеСпискаРазрешено * ● ЭтоАвторизованныйПользователь * ● ДляВсехСтрок * ● ДляОднойИзСтрок * ● Отключено * ● Кроме * ● Только | * ● AttachAdditionalTables * ● ThisList * ● AllowReadUpdate * ● AllowRead * ● AllowUpdateIfReadingAllowed * ● ValueAllowed * ● ObjectReadingAllowed * ● ObjectUpdateAllowed * ● ListReadingAllowed * ● ListUpdateAllowed * ● IsAuthorizedUser * ● ForAllRows * ● ForAtLeastOneRow * ● Disabled * ● Except * ● Only |

**Настройка свойств объектов метаданных**

Таблица 3.117.

|  |  |
| --- | --- |
| Свойство объекта метаданных | Настройка |
| ОпределяемыйТип.ЗначениеДоступа | ● Обязательные типы:  ● СправочникСсылка.Пользователи,  ● СправочникСсылка.ГруппыПользователей,  ● СправочникСсылка.ВнешниеПользователи,  ● СправочникСсылка.ГруппыВнешнихПользователей.  ● Типы значений доступа, например СправочникСсылка.Организации.  ● Типы групп значений доступа, например СправочникСсылка.\_ДемоГруппыДоступаКонтрагентов |
| ОпределяемыйТип.ВладелецНастроекПрав | ● Типы владельцев настроек прав, например СправочникСсылка.ПапкиФайлов |
| ОпределяемыйТип.ВладелецНастроекПравОбъект | ● Типы владельцев настроек прав, например СправочникОбъект.ПапкиФайлов |
| ОпределяемыйТип.ЗначениеДоступаСГруппамиЗначенийДоступаОбъект | ● Типы значений доступа, для которых указан тип групп значений (например, группы доступа контрагентов СправочникОбъект.\_ДемоКонтрагенты) |
| ОпределяемыйТип.ВладелецНаборовЗначенийДоступаОбъект (совместно с типами, указанными в создаваемых подписках на события ЗаписатьНаборыЗначенийДоступа<Окончание>) | ● Типы объектов, для которых записываются наборы значений доступа используемые, как значения в проверках по виду Объект через шаблон #ПоЗначениямИНаборамРасширенный.  ● Типы объектов с ограничением, настроенным с помощью шаблона #ПоНаборамЗначений |
| ОпределяемыйТип.ВладелецВнешнихЗначенийВНаборахЗначенийДоступаОбъект (совместно с типами, указанными в создаваемых подписках на события ЗаписатьЗависимыеНаборыЗначенийДоступа<Окончание>) | Типы объектов, которые используются в процедурах ЗаполнитьНаборыЗначенийДоступа для формирования наборов, но не входят в состав типов владельцев наборов значений доступа, т. е. отсутствуют в типе ОпределяемыйТип.ВладелецНаборовЗначенийДоступаОбъект (а также среди типов, указанных в соответствующих создаваемых подписках на события). Например, СправочникОбъект.Файлы, используемый для формирования наборов БизнесПроцессОбъект.\_ДемоЗаданиеСРолевойАдресацией (см. также обработчик УправлениеДоступомПереопределяемый.ПриИзмененииНаборовЗначенийДоступа(…)) |
| ОпределяемыйТип.ВладелецСОграничениемПоНаборамЗначенийДоступаОбъект (совместно с типами, указанными в создаваемых подписках на события ЗаполнитьНаборыЗначенийДоступаТабличныхЧастей<Окончание>) | Используется для тех объектов (кроме документов), которые используют наборы значений доступа в шаблоне #ПоНаборамЗначений для ограничения объекта. Для документов необходимо создавать отдельную подписку |
| ТабличнаяЧасть.НаборыЗначенийДоступа.(НомерНабора, ЗначениеДоступа, Уточнение, Чтение, Изменение) | Табличная часть создается в объектах метаданных прикладной конфигурации, которые используют наборы значений доступа в шаблоне #ПоНаборамЗначений для ограничения доступа к ним.  Состав полей табличной части одинаков для всех таких объектов. Состав типов, как у тех же полей в регистре сведений НаборыЗначенийДоступа |
| ОпределяемыйТип.ВладелецЗначенийКлючейДоступа | Все ссылочные объекты с ограничением доступа, а также объекты без ограничения, которые записывают ключи доступа для других объектов, например, журналов документов.  Обновляется автоматически через ПроверкаВнедренияБСП.erf |
| ОпределяемыйТип.ВладелецЗначенийКлючейДоступаОбъект | Все ссылочные объекты (кроме документов) из определяемого типа ВладелецЗначенийКлючейДоступа, а также объекты, которые участвуют в ограничениях доступа других объектов (обращение "через точку" или путем соединения).  Обновляется автоматически через ПроверкаВнедренияБСП.erf |
| ОпределяемыйТип.ВладелецЗначенийКлючейДоступаДокумент | Все документы из определяемого типа ВладелецЗначенийКлючейДоступа, а также документы, которые участвуют в ограничениях доступа других объектов (обращение "через точку" или путем соединения).  Обновляется автоматически через ПроверкаВнедренияБСП.erf |
| ОпределяемыйТип.ВладелецЗначенийКлючейДоступаНаборЗаписей | Все регистры с ограничением (кроме регистров расчета), а также регистры, которые участвуют в ограничениях доступа других объектов путем соединения.  Обновляется автоматически через ПроверкаВнедренияБСП.erf. |
| ОпределяемыйТип.ВладелецЗначенийКлючейДоступаНаборЗаписейРегистраРасчета | Все регистры расчета с ограничением, а также регистры расчета, которые участвуют в ограничениях доступа других объектов путем соединения.  Обновляется автоматически через ПроверкаВнедренияБСП.erf. |
| ОпределяемыйТип.ПолеРегистраКлючейДоступаКРегистрам | Все типы собственных полей регистра, указанных в ограничении доступа к ним (типы опорных полей - аналог ссылки у объекта).  Обновляется автоматически через ПроверкаВнедренияБСП.erf. |
| РегистрСведений.КлючиДоступаКРегистру<ИмяРегистраСОграничениемДоступа> | Отдельный регистр сведений для хранения ключей доступа целевого регистра рекомендуется создавать вместо использования общего поставляемого регистра КлючиДоступаКРегистрам, если в целевом регистре комбинаций значений полей, которые используются в ограничении доступа, может быть очень много (десятки тысяч и более).  Для создания отдельного регистра нужно скопировать поставляемый регистр сведений КлючиДоступаКРегистрам и переименовать. После этого лишние измерения Поле\* удалить, а состав типов оставшихяся измерений Поле\* указать одинаковый, полученный как сумма типов всех полей целевого регистра, которые используются в ограничении доступа. |

**Описание видов доступа, используемых в ограничениях объектов для отчета «Права доступа»**

Для использования отчета Права доступа требуется задать список видов доступа, используемых в ограничении прав объектов метаданных. Список задается в переопределяемом модуле в виде многострочной строки формата <Полное имя таблиц>.<Право>.<Вид доступа>. Для автоматического формирования кода процедуры следует применить инструменты разработчика. Пример заполнения процедуры:

Процедура ПриЗаполненииВидовОграниченийПравОбъектовМетаданных(Описание) Экспорт

Описание =

"

|БизнесПроцесс.Задание.Добавление.Пользователи

|БизнесПроцесс.Задание.Изменение.Пользователи

|БизнесПроцесс.Задание.Чтение.Пользователи

|…

|…

|…

|Справочник.ПапкиФайлов.Изменение.\_ДемоПапкиФайлов

|Справочник.ПапкиФайлов.Чтение.\_ДемоПапкиФайлов

|Справочник.УчетныеЗаписиЭлектроннойПочты.Чтение.УчетныеЗаписиЭлектроннойПочты

|";

КонецПроцедуры

**Создание описаний поставляемых профилей групп доступа для начального заполнения и обновления информационной базы**

Для создания описаний профилей групп доступа, поставляемых в конфигурации, используемых при начальном заполнении и обновлении информационной базы, а также при интерактивном восстановлении поставляемых профилей, измененных администратором системы, используется процедура ЗаполнитьПоставляемыеПрофилиГруппДоступа общего модуля УправлениеДоступомПереопределяемый.

Для создания кода описания на встроенном языке рекомендуется применять обработку в демонстрационной конфигурации УправлениеДоступом.

Процедура ЗаполнитьПоставляемыеПрофилиГруппДоступа(ОписанияПрофилей, ПараметрыОбновления) Экспорт

// Профиль "Пользователь".

ЗаполнитьПрофильПользователь(ОписанияПрофилей);

// Профиль "Бухгалтер".

ЗаполнитьПрофильБухгалтер(ОписанияПрофилей);

...

// Дополнительные профили, которые не используются самостоятельно при настройке

// прав пользователя, а дополняют основные профили, перечисленные выше.

// Профиль "Ответственный за составы участников групп доступа (дополнительно)".

ЗаполнитьПрофильОтветственныйЗаСоставыУчастниковГруппДоступа(ОписанияПрофилей);

// Профиль "Ответственный за ведение номенклатуры (дополнительно)".

ЗаполнитьПрофильОтветственныйЗаВедениеНоменклатуры(ОписанияПрофилей);

...

КонецПроцедуры

Процедура ЗаполнитьПрофильПользователь(Знач ОписанияПрофилей)

ОписаниеПрофиля = УправлениеДоступом.НовоеОписаниеПрофиляГруппДоступа();

ОписаниеПрофиля.Имя           = "\_ДемоПользователь";

ОписаниеПрофиля.Идентификатор = "09e56dbf-90a0-11de-862c-001d600d9ad2";

ОписаниеПрофиля.Наименование  = НСтр("ru = 'Пользователь'");

ОписаниеПрофиля.Описание =

НСтр("ru = 'Общие разрешенные действия для большинства пользователей.

|Как правило, это права на просмотр данных информационной системы.'");

// Использование 1С:Предприятия.

ОписаниеПрофиля.Роли.Добавить("ЗапускВебКлиента");

ОписаниеПрофиля.Роли.Добавить("ЗапускТолстогоКлиента");

ОписаниеПрофиля.Роли.Добавить("ЗапускТонкогоКлиента");

ОписаниеПрофиля.Роли.Добавить("ВыводНаПринтерФайлБуферОбмена");

ОписаниеПрофиля.Роли.Добавить("СохранениеДанныхПользователя");

// Использование программы.

ОписаниеПрофиля.Роли.Добавить("БазовыеПраваБСП");

ОписаниеПрофиля.Роли.Добавить("ПросмотрОписанияИзмененийПрограммы");

...

ОписанияПрофилей.Добавить(ОписаниеПрофиля);

КонецПроцедуры

...

Внимание. Для того чтобы изменения вступили в силу сразу при разработке и отладке конфигурации, требуется обновить вспомогательные данные. См. раздел «[Обновление вспомогательных данных во время разработки](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_обновление_вспомогательных_данных)».

**Размещение в командном интерфейсе**

Если в конфигурации не используется подсистема «Настройки программы», то в командном интерфейсе необходимо разместить:

* ● Справочник.ГруппыДоступа (кроме случая упрощенного интерфейса):
* ● для частой работы администратора по просмотру и редактированию групп доступа (например, в подразделе Важное рядом со справочниками Пользователи и ВнешниеПользователи);
* ● для частой работы администратора группы доступа по просмотру и редактированию участников групп доступа (например, в подразделе Важное рядом со справочником Пользователи).
* ● Справочник.ПрофилиГруппДоступа (кроме случая упрощенного интерфейса) – для редкой работы администратора по просмотру списка и содержания профилей групп доступа (например, в подразделе См. также).
* ● ОграничиватьДоступНаУровнеЗаписей – для включения/выключения ограничения доступа на уровне записей (администраторами). Если в ограничениях прав используются наборы значений доступа, то при включении константы необходимо выдавать предупреждение о необходимости заполнения данных для ограничения доступа.

См. пример в форме НастройкиПользователейИПрав обработки ПанельАдминистрированияБСП.

**Настройка прав доступа пользователей**

Для настройки прав доступа пользователей к данным подсистемы «Управление доступом» следует использовать следующие роли:

Таблица 3.118.

| № | Роли и их назначение |
| --- | --- |
| 1. | БазовыеПраваБСП (из подсистемы «Базовая функциональность»)  Просмотр списка пользователей.  Просмотр списка своих групп доступа в форме Права доступа.  Формирование и просмотр отчета о своих правах доступа |
| 2. | ИзменениеУчастниковГруппДоступа  Изменение состава участников и просмотр тех групп доступа, у которых пользователь установлен как администратор группы доступа |
| 3. | ПолныеПрава (из подсистемы «Базовая функциональность»)  Просмотр и редактирование групп доступа (в том числе назначение администраторов групп доступа).  Удаление объектов подсистемы, помеченных на удаление.  Ручной запуск регламентного задания Заполнение данных для ограничения доступа.  Использование обработки Обновление доступа на уровне записей.  Включение и отключение ограничений прав доступа на уровне записей.  Разработка профилей групп доступа |

Пример настройки прав доступа пользователей:

| № | Группа пользователей и ее функции | Состав ролей |
| --- | --- | --- |
|  | Ответственный за составы участников групп доступа (дополнительно):  ● Нужен для работы пользователя, которого назначили ответственным за состав участников в группе доступа (например, начальник подразделения, руководитель проектной группы и др.) | ● ИзменениеУчастниковГруппДоступа,  ● <Подсистема\_ДемоАдмиристрирование> (просмотр подсистемы Администрирование подсистемы «Настройки программы») |

**Особые случаи внедрения подсистемы**

* ● [Внедрение подсистем «Бизнес-процессы и задачи» и «Управление доступом»](https://its.1c.ru/db/content/bsp301doc/src/3.63.%20особые%20случаи%20внедрения%20подсистем.htm?_=1542292879#_внедрение_подсистем_«бизнес-процесс);
* ● [Внедрение подсистем «Взаимодействия» и «Управление доступом»](https://its.1c.ru/db/content/bsp301doc/src/3.63.%20особые%20случаи%20внедрения%20подсистем.htm?_=1542292879#_внедрение_подсистем_«взаимодействия);
* ● [Внедрение подсистем «Контроль ведения учета» и «Управление доступом»](https://its.1c.ru/db/content/bsp301doc/src/3.63.%20особые%20случаи%20внедрения%20подсистем.htm?_=1542292879#_внедрение_подсистем_«контроль);
* ● [Внедрение подсистем «Работа с почтовыми сообщениями» и «Управление доступом»](https://its.1c.ru/db/content/bsp301doc/src/3.63.%20особые%20случаи%20внедрения%20подсистем.htm?_=1542292879#_внедрение_подсистем_«работа);
* ● [Внедрение подсистем «Работа с файлами» и «Управление доступом»](https://its.1c.ru/db/content/bsp301doc/src/3.63.%20особые%20случаи%20внедрения%20подсистем.htm?_=1542292879#_внедрение_подсистем_«работа_1);
* ● [Внедрение подсистем «Свойства» и «Управление доступом»](https://its.1c.ru/db/content/bsp301doc/src/3.63.%20особые%20случаи%20внедрения%20подсистем.htm?_=1542292879#_внедрение_подсистем_«свойства»_1).

**Использование при разработке конфигурации**

При создании новых прикладных объектов метаданных нужно повторно выполнять соответствующую часть процедуры настройки, описанной выше.

Программный интерфейс подсистемы описан в соответствующем разделе главы 4 «[Программный интерфейс](https://its.1c.ru/db/content/bsp301doc/src/глава%204.%20программный%20интерфейс.htm?_=1542292879)».

**Обеспечение производительности пакетной обработки данных (полноправный пользователь)**

Когда константа ОграничиватьДоступаНаУровенЗаписейУнивервально включена, тогда при изменении объектов (или наборов записей), в том числе, в режиме ОбменДанными.Загрузка = Истина, происходит обновление ключей доступа к объектам (или к регистрам), которое необходимо для корректной проверки прав доступа для неполноправных пользователей. Это добавляет, как правило, 5-20 мс на 1 объект или набор записей, что при записи большого числа "легких" объектов может быть существеннным замедлением. Чтобы сократить эти затраты, следует отключить обновление ключей доступа на период загрузки, а затем включить (при включении произойдет планирование обновления ключей доступа для таблиц, в которые внесены изменения с момента отключения). Таким образом, общие затраты на загрузку будут снижены, а обновление ключей доступа будет выполнено в фоне, что более эффективно, если объектов много, так как в фоне обновление ключей доступа выполняется пакетно, а не по одному. Для отключения обновления ключей доступа предусмотрена процедура ОтключитьОбновлениеКлючейДоступа общего модуля УправлениеДоступом (примеры см. в комментарии к процедуре).

**Проверка работы логики ограничений прав**

Ошибка доступа к данным связана либо с недостаточным, либо избыточным доступом к данным информационной базы. В общем виде подобные нештатные ситуации могут возникать по причинам, перечисленным ниже.

* ● Возможная ошибка при настройке прав доступа:
* ● некорректная настройка состава пользователей или групп пользователей группы доступа – необходимо включить пользователя или группу пользователей в группы доступа или исключить его из них;
* ● некорректный профиль группы доступа – выбрать другой профиль групп доступа, создать дополнительный профиль или скорректировать имеющийся профиль;
* ● некорректная настройка ограничений доступа в группе доступа (списков разрешенных значений, например: организаций, групп партнеров, складов) – необходимо добавить или исключить значений доступа из группы доступа.
* ● Возможная ошибка при разработке или доработке конфигурации на внедрении (в компетенции разработчика конфигурации):
* ● некорректная настройка состава ролей в профиле;
* ● некорректная настройка состава видов доступа в профиле.
* ● Ошибки разработчика конфигурации:
* ● ошибка в параметрах стандартного шаблона;
* ● ошибка в составе типов подписок на события;
* ● ошибка в процедуре заполнения наборов значений доступа;
* ● ошибка в описании видов доступа.
* ● Ошибки разработчика этой подсистемы:
* ● ошибка в стандартном шаблоне;
* ● ошибка особенностей работы СУБД;
* ● ошибка в общих модулях подсистемы.

Для диагностики ошибок доступа к данным можно воспользоваться следующими средствами:

* ● Убедиться, что содержимое регистра сведений Наборы значений доступа содержит наборы значений доступа, которые записаны таким образом, что наборы с одинаковыми номерами для одного объекта соответствуют логике ограничения доступа по условию И, а затем наборы с разными номерами «складываются» по условию ИЛИ:
* ● если по интересующему нас объекту в регистре сведений нет записей, то объект не будет доступен для всех пользователей;
* ● если по интересующему нас объекту нет ни одного набора, относящегося к конкретному праву доступа, значит, для всех пользователей нет этого права на объект.
* ● Нужно иметь в виду, что если в профиль не добавлена роль для чтения каких-либо данных, но предусмотрены роли для работы с другими связанными объектами, то возможна ситуация, когда в полях этих объектов будут отображаться значения вида Ссылка не найдена (… : …) вместо самих данных.

**Разработка обработчиков обновления ИБ**

В процессе разработки конфигурации в нее могут быть внесены изменения, которые потребуют обновления данных для ограничения доступа. Ниже перечислены типовые ситуации, когда обновление необходимо.

1. При каждом обновлении информационной базы выполняются обработчики обновления в модуле УправлениеДоступомСлужебный:

* ● Для обновления неразделенных вспомогательных данных (кэшей свойств встраивания), которые сохраняются в регистре сведений ПараметрыРаботыПрограммы вызывается процедура ОбновитьПараметрыОграниченияДоступа. Процедура обновляет общие параметры, права ролей, зависимости прав доступа, описания поставляемых профилей, состав предопределенных профилей, описание возможных прав для настройки прав объектов, например:
* ● при добавлении объектов метаданных (доступ к которым может ограничиваться) и добавлении/изменении ролей обновляется регистр сведений ПраваРолей и регистрируется состав объектов метаданных, для которых внесены изменения;
* ● при изменении описаний поставляемых профилей групп доступа регистрируются их изменения;
* ● при изменении зависимостей прав доступа между объектами метаданных регистрируются их изменения.
* ● Для обновления разделенных вспомогательных данных (разных для каждой области данных) вызывается процедура ОбновитьВспомогательныеДанныеПоИзменениямКонфигурации. В том числе обновляется регистр сведений ПараметрыОграниченияДоступа.

Соответственно, в процессе разработки необходимо учитывать, что для «вступления в силу» перечисленных изменений необходим запуск обработчиков (перед проверкой текущей разработки). При этом, если был изменен только текст ограничения в модуле менеджера, то при попытке изменить этот объект подсистема автоматически обновит параметры ограничения доступа.

Обычно для обновления требуется изменить версию конфигурации на следующую. Для повторных запусков обработчиков следует изменять версию в регистре сведений ВерсииПодсистем на предыдущую.

Внимание. В процессе разработки рекомендуется использовать обработку ОбновлениеВспомогательныхДанных.epf, которая входит в состав дистрибутива библиотеки. При закладке в хранилище результатов разработки, требующих [обновления вспомогательных данных](https://its.1c.ru/db/content/bsp301doc/src/3.4.%20базовая%20функциональность.htm?_=1542292879#_обновление_вспомогательных_данных), рекомендуется поместить в хранилище новую версию конфигурации, чтобы обновление информационной базы выполнилось у всех участков разработки.

2. Обновление поставляемых профилей групп доступа. Если в конфигурации было изменено описание поставляемых профилей групп доступа в процедуре ПриЗаполненииПоставляемыхПрофилейГруппДоступа общего модуля УправлениеДоступомПереопределяемый, то никаких действий не требуется. При обновлении информационной базы обновление поставляемых профилей будет выполнено автоматически. Для управления обновлением предусмотрены свойства в параметре процедуры ПараметрыОбновления. Их значения можно переопределить (подробнее см. комментарий к процедуре).

3. Начальное заполнение наборов значений доступа необходимо выполнить при включении ограничений на уровне записей, если в ролях конфигурации имеются ограничения прав, которые используют наборы значений доступа.

При этом, если среди видов доступа применяются группы значений доступа, также требуется выполнить заполнение групп значений доступа.

Для выполнения начального заполнения следует воспользоваться одним из двух способов:

* ● Разрешить выполнение регламентного задания ЗаполнениеДанныхДляОграниченияДоступа; этот способ позволяет выполнить начальное заполнение наборов значений доступа порциями в фоне, не мешая работе администратора. Однако для того, чтобы изменения вступили в силу и с базой смогли начать работу другие пользователи, необходимо дождаться завершения отработки регламентным заданием всех порций.
* ● Вызвать процедуру ЗаполнениеДанныхДляОграниченияДоступа общего модуля УправлениеДоступом с параметром КоличествоДанных = 0\*.

4. Обновление наборов значений доступа необходимо выполнять в случаях:

* ● Если в конфигурации были добавлены виды доступа, использующие группы значений доступа, или изменены существующие виды доступа так, что они стали использовать группы значений доступа, то необходимо выполнить обновление наборов значений доступа. Для этого следует воспользоваться методикой, описанной в пункте 3.
* ● Если в конфигурации у определенных объектов метаданных был изменен алгоритм формирования наборов значений доступа или были изменены зависимости по правам. Для решения этой задачи следует выбрать один из двух вариантов:
* ● перезаписать наборы значений доступа с помощью процедуры ОбновитьНаборыЗначенийДоступа общего модуля УправлениеДоступом;
* ● очистить наборы значений доступа для соответствующих объектов и заполнить их способом, который описан в пункте 3.

\* Операция обновления наборов значений доступа соизмерима с перезаписью всех документов, доступ к которым ограничивается на уровне записей, поэтому процесс обновления на большой базе может занять достаточно много времени. Поэтому лучше использовать предусмотренное регламентное задание. Для этого достаточно включить его использование (отключение произойдет автоматически).

5. Обновление ключей доступа выполняется автоматически. Ход обновления можно проконтролировать в форме Обновление доступа на уровне записей (раздел Администрирование - Настройки пользователей и прав).

**Настройка обмена данными**

Для настройки обмена данными следует руководствоваться общими правилами.

В планы обмена распределенной информационной базы (РИБ) без фильтра рекомендуется включать все объекты метаданных подсистемы.

В планы обмена распределенной информационной базы (РИБ) с фильтром рекомендуется включать все объекты метаданных подсистемы, кроме:

* ● справочник КлючиДоступа,
* ● справочник НаборыГруппДоступа,
* ● регистр сведений КлючиДоступаКОбъектам,
* ● регистр сведений КлючиДоступаКРегистрам,
* ● регистры сведений КлючиДоступаКРегистру<\*>,
* ● регистр сведений КлючиДоступаГруппДоступа,
* ● регистр сведений КлючиДоступаНаборовГруппДоступа,
* ● регистр сведений КлючиДоступаПользователей,
* ● регистр сведений КлючиДоступаВнешнихПользователей,
* ● регистр сведений ОбновлениеКлючейДоступаКДанным,
* ● регистр сведений ОбновлениеКлючейДоступаПользователей.

В планах обмена распределенной информационной базы (РИБ) рекомендуется отключать регистрацию изменений для следующих объектов метаданных подсистемы (см. также раздел «[Особенности создания начального образа подчиненного узла распределенной ИБ](https://its.1c.ru/db/content/bsp301doc/src/3.30.%20обмен%20данными.htm?_=1542292879#_особенности_создания_начального)»):

* ● константа КоличествоПотоковОбновленияДоступа,
* ● константа ПоследнееОбновлениеДоступа,
* ● справочник КлючиДоступа,
* ● справочник НаборыГруппДоступа,
* ● регистр сведений КлючиДоступаКОбъектам,
* ● регистр сведений КлючиДоступаКРегистрам,
* ● регистры сведений КлючиДоступаКРегистру<\*>,
* ● регистр сведений КлючиДоступаГруппДоступа,
* ● регистр сведений КлючиДоступаНаборовГруппДоступа,
* ● регистр сведений КлючиДоступаПользователей,
* ● регистр сведений КлючиДоступаВнешнихПользователей,
* ● регистр сведений ОбновлениеКлючейДоступаКДанным,
* ● регистр сведений ОбновлениеКлючейДоступаПользователей,
* ● регистр сведений ПараметрыОграниченияДоступа,
* ● регистр сведений ПраваРолей,
* ● регистр сведений ЗависимостиПравДоступа,
* ● регистр сведений ТаблицыГруппДоступа,
* ● регистр сведений ЗначенияГруппДоступа,
* ● регистр сведений ЗначенияГруппДоступаПоУмолчанию.

Из остальных планов обмена (не РИБ) рекомендуется исключать следующие объекты метаданных:

* ● константа КоличествоПотоковОбновленияДоступа,
* ● константа ПоследнееОбновлениеДоступа,
* ● справочник КлючиДоступа,
* ● справочник НаборыГруппДоступа,
* ● регистр сведений КлючиДоступаКОбъектам,
* ● регистр сведений КлючиДоступаКРегистрам,
* ● регистры сведений КлючиДоступаКРегистру<\*>,
* ● регистр сведений КлючиДоступаГруппДоступа,
* ● регистр сведений КлючиДоступаНаборовГруппДоступа,
* ● регистр сведений КлючиДоступаПользователей,
* ● регистр сведений КлючиДоступаВнешнихПользователей,
* ● регистр сведений ОбновлениеКлючейДоступаКДанным,
* ● регистр сведений ОбновлениеКлючейДоступаПользователей,
* ● регистр сведений ПараметрыОграниченияДоступа,
* ● регистр сведений ПраваРолей,
* ● регистр сведений ЗависимостиПравДоступа,
* ● регистр сведений ТаблицыГруппДоступа,
* ● регистр сведений ЗначенияГруппДоступа,
* ● регистр сведений ЗначенияГруппДоступаПоУмолчанию.