**Сравнение баз данных сигналов проектов**

*Описание функции сравнения содержимого двух баз данных SDB.*

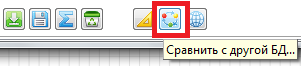
Для организации коллективной работы пользователей с одним проектом в среде SimInTech предусмотрена функция автоматического сравнения созданных/измененных баз данных (БД). Данная функция позволяет корректно объединять в одну БД несколько версий БД, созданных разными пользователями, а также отследить во времени изменения, внесенные пользователем в текущую БД проекта.

Даная функция позволяет выполнить сравнительный анализ созданных баз данных проектов по следующим критериям:

* поиск новых категорий сигналов, не существующих в текущей БД проекта,
* поиск удаленных категорий из текущей базы данных,
* поиск новых групп сигналов в рамках каждой категории, не существующих в текущей базе данных проекта,
* поиск изменений, внесенных в шаблоны категорий,
* поиск групп сигналов, удаленных из текущей базы данных,
* поиск новых сигналов, не существующих в текущей базе данных проекта,
* поиск сигналов, удаленных из текущей базы данных,
* поиск сигналов, не совпадающих по свойствам (различается тип данных, режим расчета либо начальное значение).

Для выполнения сравнения БД необходимо выполнить следующие действия:

1) В правом нижнем углу редактора БД выбрать кнопку **«Сравнить с другой БД…»**.



**Рисунок 1. Кнопка сравнения БД проекта**

Файл выбранной для сравнения БД загружается в память временно, только на период выполнения сравнения, и не связан с prt-файлом.

2) При загрузке второй БД (БД2) выполняется взаимное сравнение ее содержимого с текущей базой данных проекта.

3) Результат сравнения отображается на двух формах, которые соответствуют по составу текущей и сравниваемой БД2 с соответствующей цветовой индикацией, в зависимости от обнаруженных отличий:

- зеленым цветом отображаются категории/группы сигналов/сигналы, полностью совпадающие по составу и свойствам,

- желтым цветом отображаются категории/группы сигналов/сигналы, совпадающие по имени, но несовпадающие по свойствам,

- красным цветом отображаются категории/группы сигналов/сигналы, не существующие в сравниваемой БД либо не совпадающие по составу,

- красными рамками выделяются отличающиеся свойства сигналов, существующих в обоих БД (имена сигналов совпадают, а название, тип данных, значение и/или способ расчета отличаются).

После выполненного сравнения и анализа результатов сравнения рекомендуется закрыть форму, соответствующую БД2. При этом в дальнейшей работе с проектом цветовая индикация БД будет сохраняться до перезагрузки проекта либо до получения новых данных сравнения.

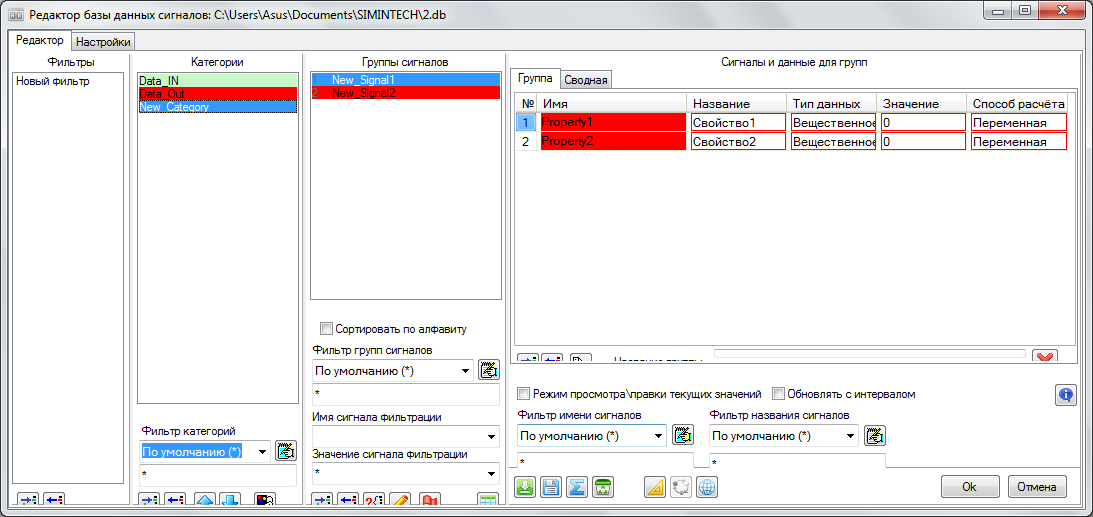
Для принудительного сброса цветовой индикации предусмотрена кнопка "Очистить результаты сравнения с другой БД" в правом нижнем углу редактора БД.



**Рисунок 2. Кнопка очистки цветовой индикации результатов сравнения БД проектов**

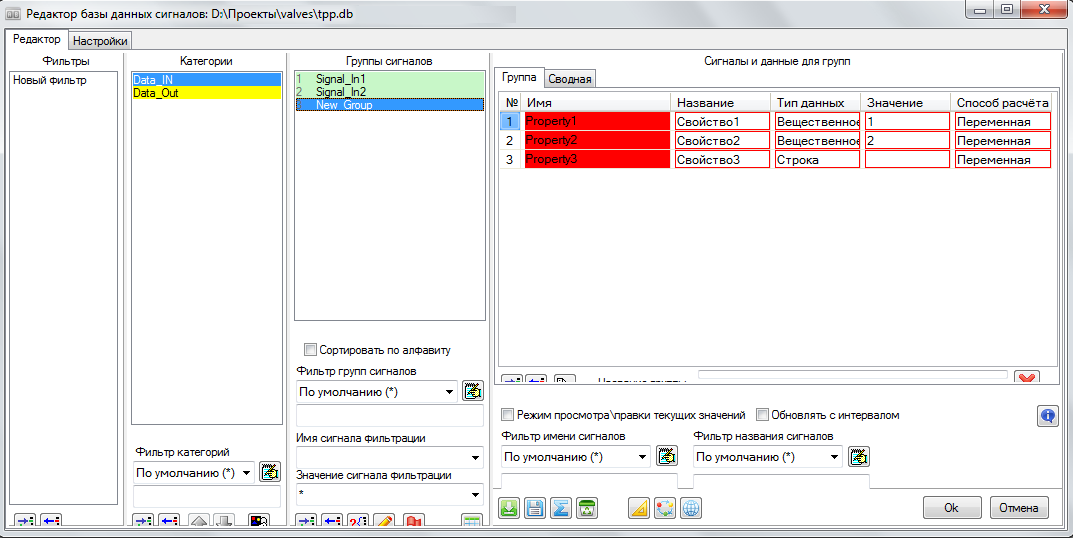
Рассмотрим несколько примеров работы функции. На рисунках 3-7 представлены результаты сравнения текущей БД проекта (далее БД1) и новой БД (далее БД2) в случаях внесения разных изменений в БД1/БД2.

* В случае добавления в БД1 новой категории данная категория, а также все ее содержимое, выделяется красным цветом (в форме сравнения БД1 с БД2). В случае если категория была удалена из БД1, данная категория также будет выделена красным цветом в форме сравнения БД2 с БД1.

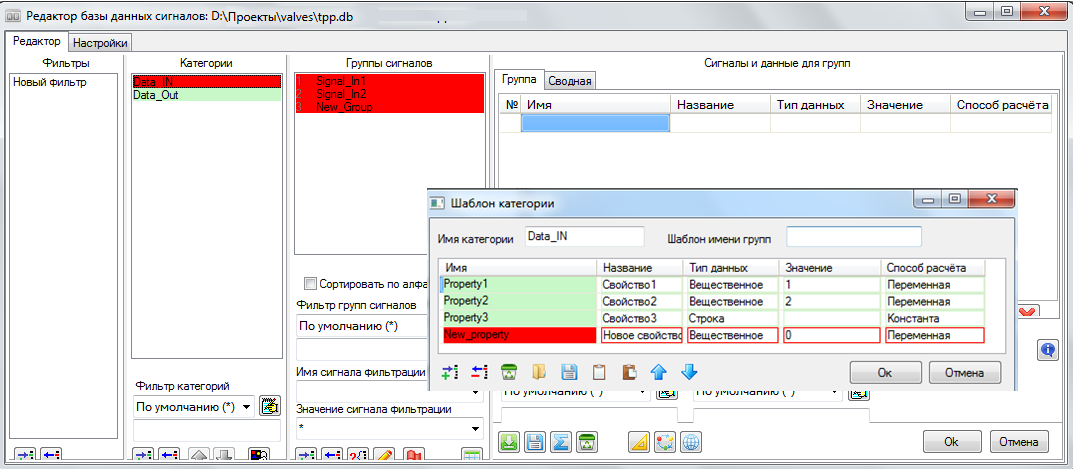
****

**Рисунок 3. Результаты сравнения в случае добавления новой категории сигналов в БД**

* В случае добавления в БД1 новой группы сигналов данная группа сигналов, выделяется красным цветом (в форме сравнения БД1 с БД2). При этом категория, в которую была добавлена новая группа, выделяется красным цветом. В случае если группа была удалена из БД1, аналогичные цветовые изменения будут отображены в форме сравнения БД2 с БД1.

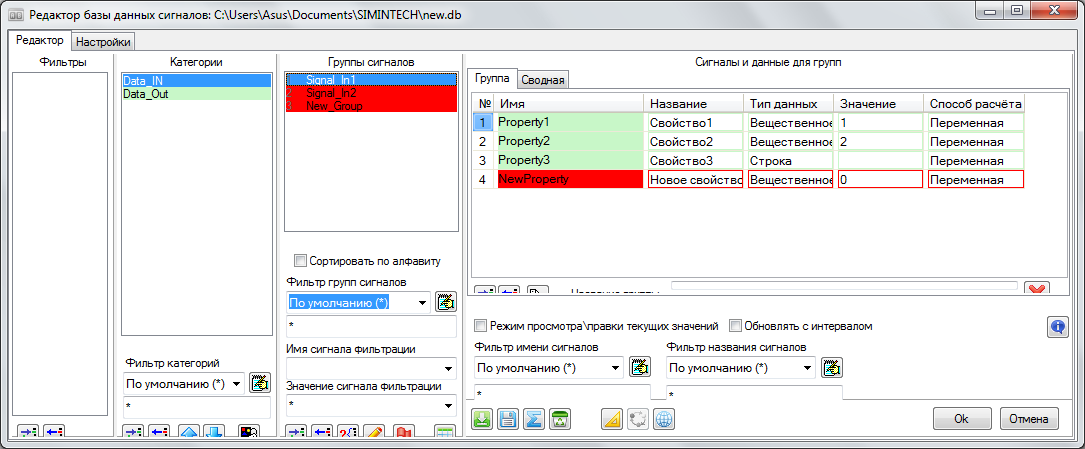
**Рисунок 4. Результаты сравнения в случае добавления новой группы сигналов в БД**

* В случае внесения изменений в шаблон существующей категории, данная категория выделяется красным цветом. При этом в шаблоне категории внесенные изменения (новые шаблонные сигналы, которых нет в сравниваемой БД) также будут выделены красным цветом.

****

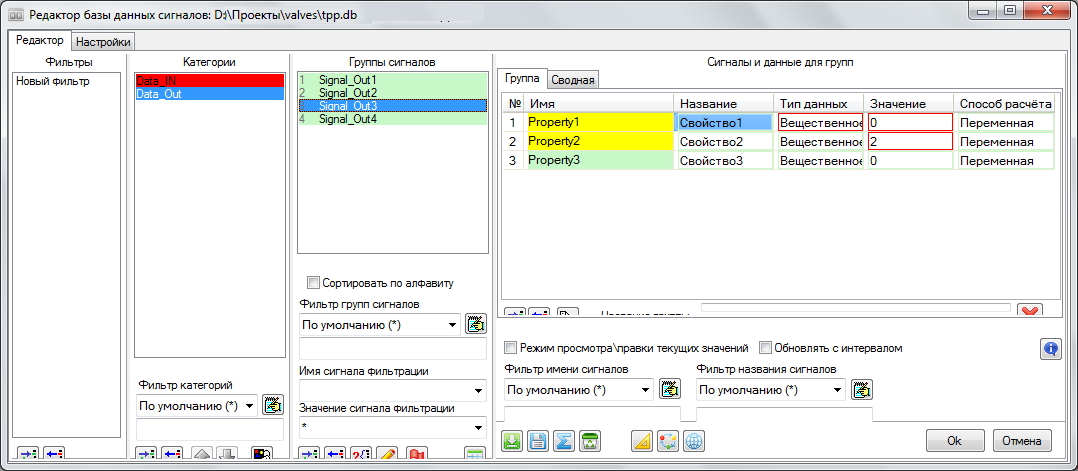
**Рисунок5. Результаты сравнения в случае изменения шаблона группы сигналов в БД**

* В случае добавления в БД1 новых сигналов в существующую группу сигналов, данные сигналы, а также группа и категория, в которую добавлены сигналы, выделяются красным цветом (в форме сравнения БД1 с БД2). В случае если сигналы были удалены из БД1, но остались в БД2, они будут выделены красным цветом в форме сравнения БД2 с БД1.

****

**Рисунок 6. Результаты сравнения в случае добавления сигналов в БД**

* В случае изменения свойств существующих сигналов (различается тип данных, режим расчета либо начальное значение) данные сигналы, а также группа сигналов и категория выделяются желтым цветом. При этом измененные свойства сигналов выделяются красной рамкой.

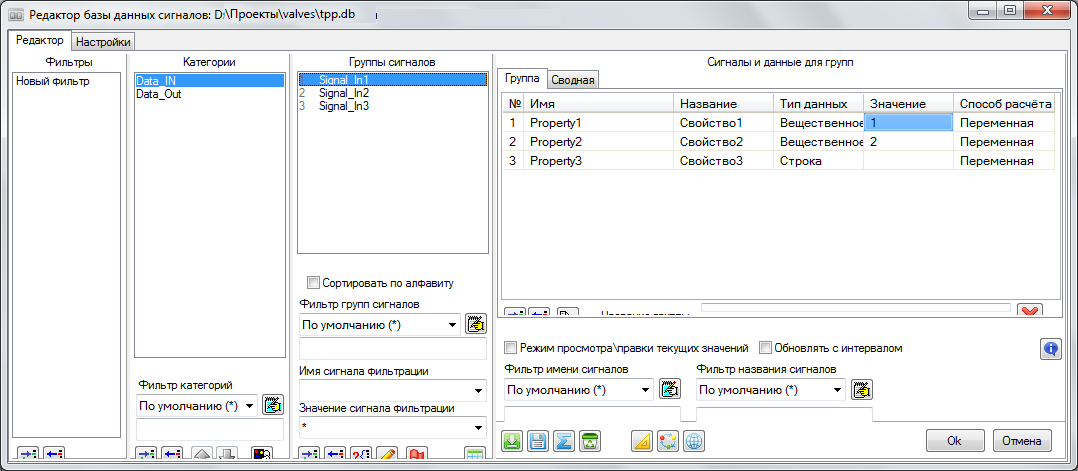
****

**Рисунок 7. Результаты сравнения в случае изменения свойств сигналов в БД**

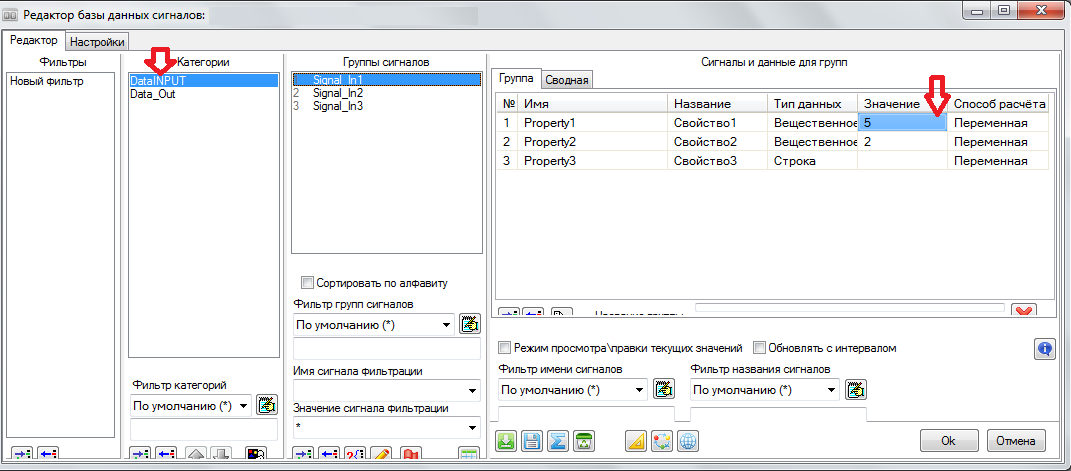
**Примеры работы с функцией сравнения БД**

Пример 1

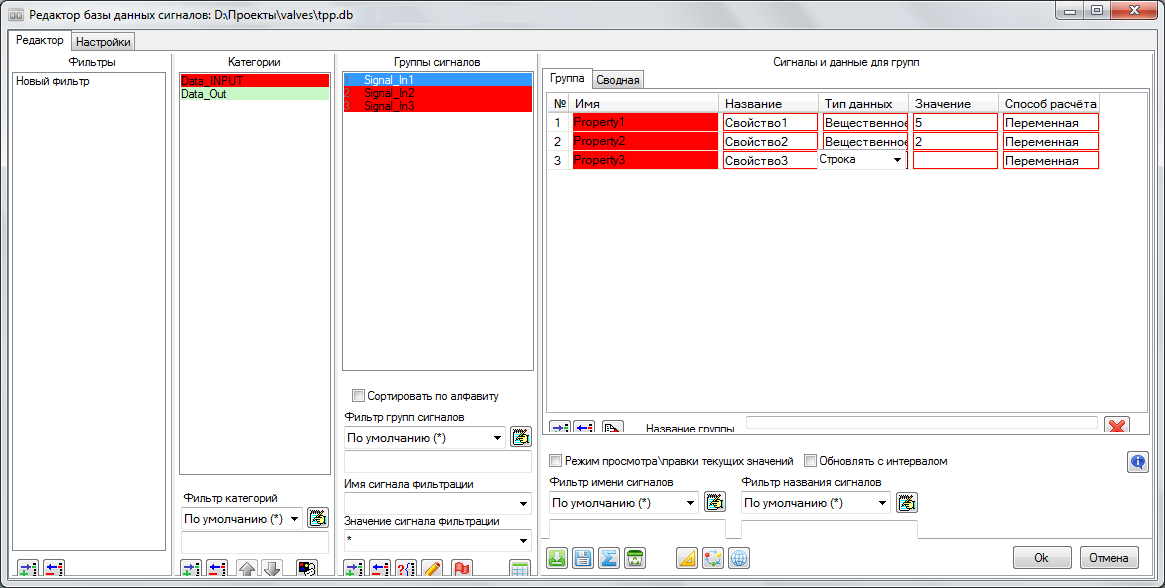
При работе пользователей с проектом одним из них была переименована категория Data\_IN в категорию DataINPUT. При этом в самой категории было изменено только начальное значение сигнала Signal\_In1\_Property1. Для объединения двух баз данных в одну необходимо предварительно выполнить сравнение двух баз данных. Результат сравнения БД1 (см. рисунок 8) и БД2 (см. рисунок 9) показан на рисунке 10.

****

**Рисунок 8 Редактор базы данных пользователя 1 (БД1)**

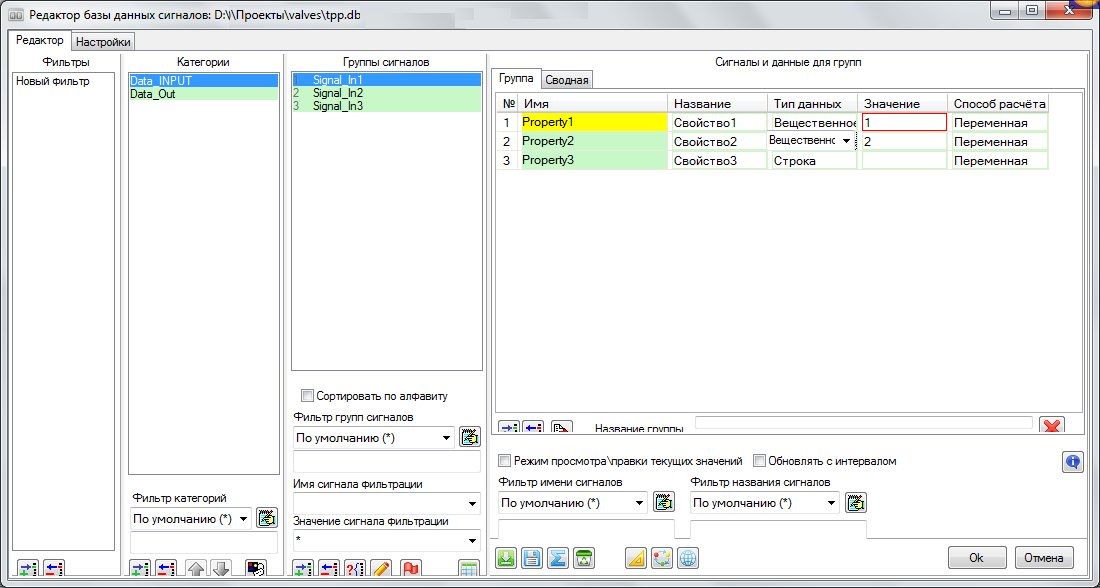
****

**Рисунок 9 Редактор базы данных пользователя 2 (БД2)**

****

**Рисунок 10 Результат сравнения БД1 и БД2 в случае несовпадения имен категорий**

Как видно на рисунке 10, все содержимое категории с измененным именем будет выделено красным цветом т.к. функция сравнения распознает ее как новую категорию, независимо от того, что содержимое категорий практически совпадает. Для продолжения работы по объединению БД необходимо исправить имя категории и заново выполнить сравнение. После повторного сравнения БД функция обнаружит измененное значение сигнала Signal\_In1\_Property1 (см. рисунок 11).

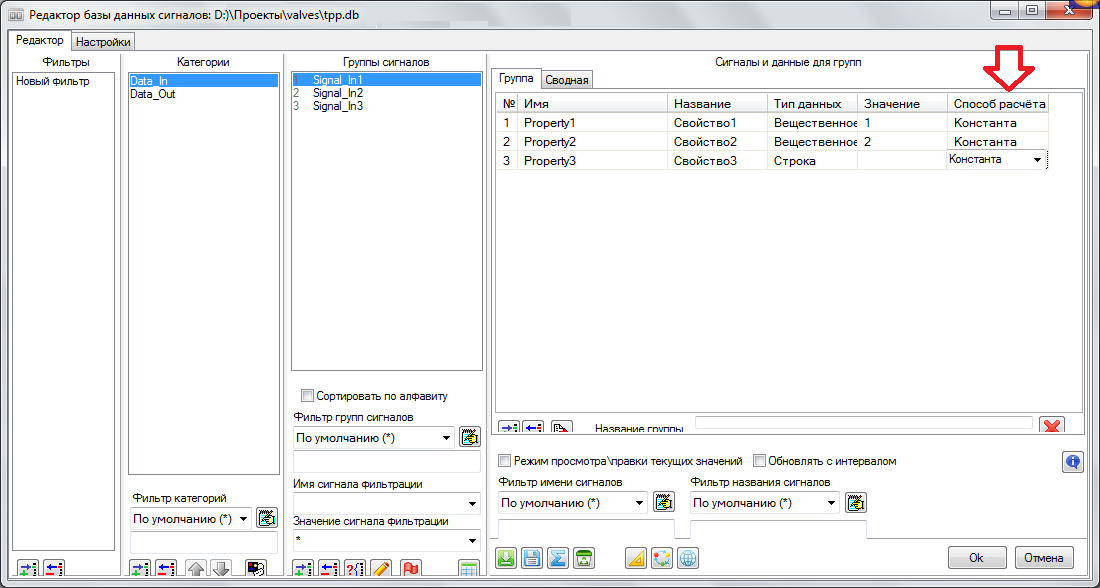


**Рисунок 11 Результат сравнения БД1 и БД2 в случае несовпадения начального значения сигнала Signal\_In1\_Property1**

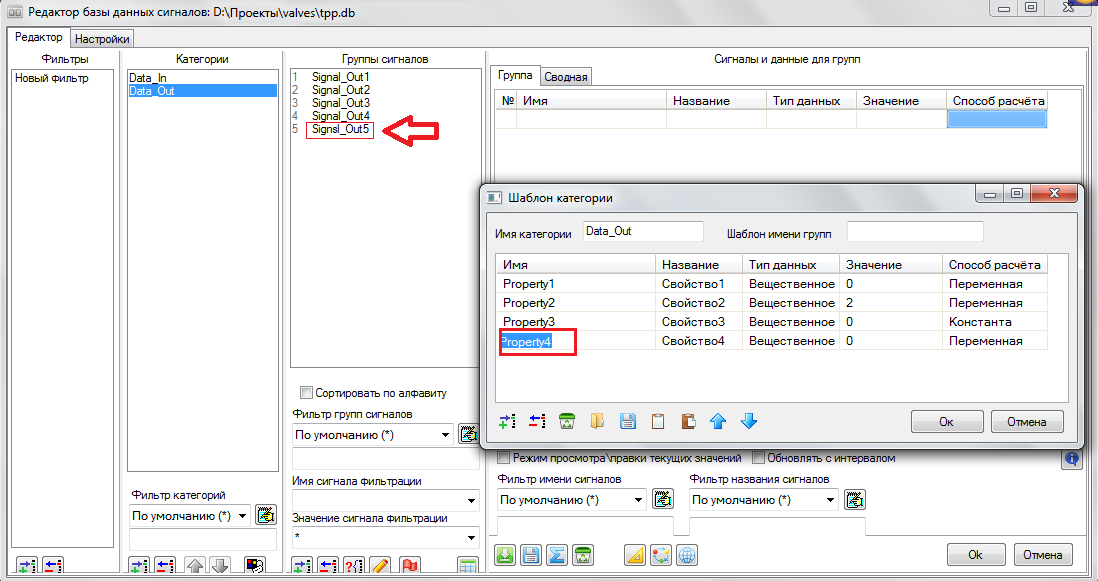
Пример 2

При работе пользователей с проектом пользователем 1 было исправлено содержимое категории Data\_In (исправлены типы данных входящих сигналов). Пользователем 2 была исправлена категория Data\_Out (добавлена новая группа сигналов Signal\_Out5, изменен шаблон категории). В итоговой БД необходимо объединить исправления двух пользователей.

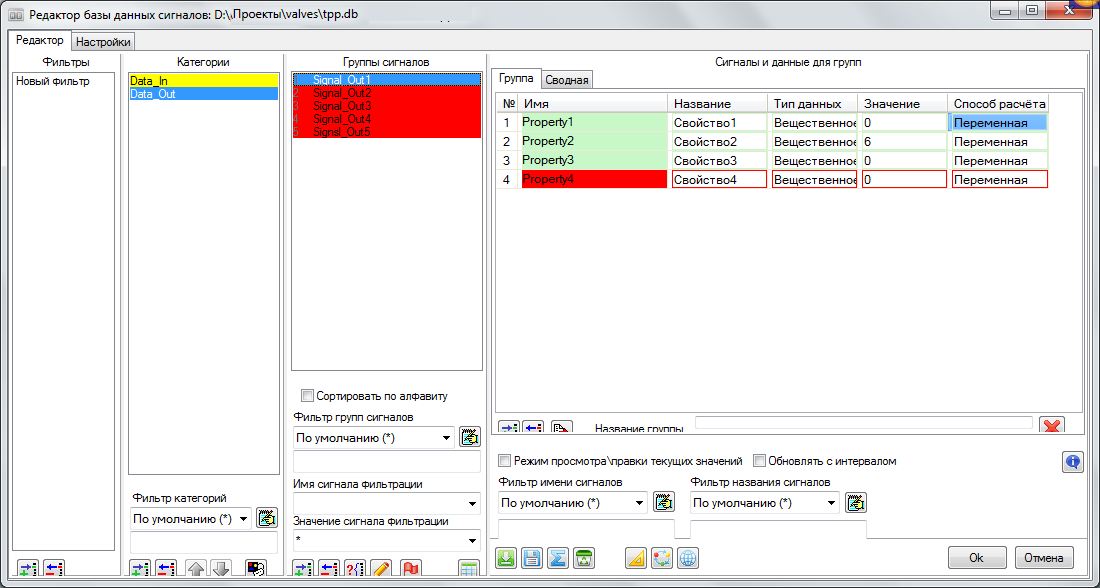
Для объединения двух БД в одну необходимо для начала выполнить их сравнение. Результат сравнения БД1 (см. рисунок 12) и БД2 (см. рисунок 13) показан на рисунке 14.



**Рисунок 12 Редактор базы данных пользователя 1 (БД1)**

****

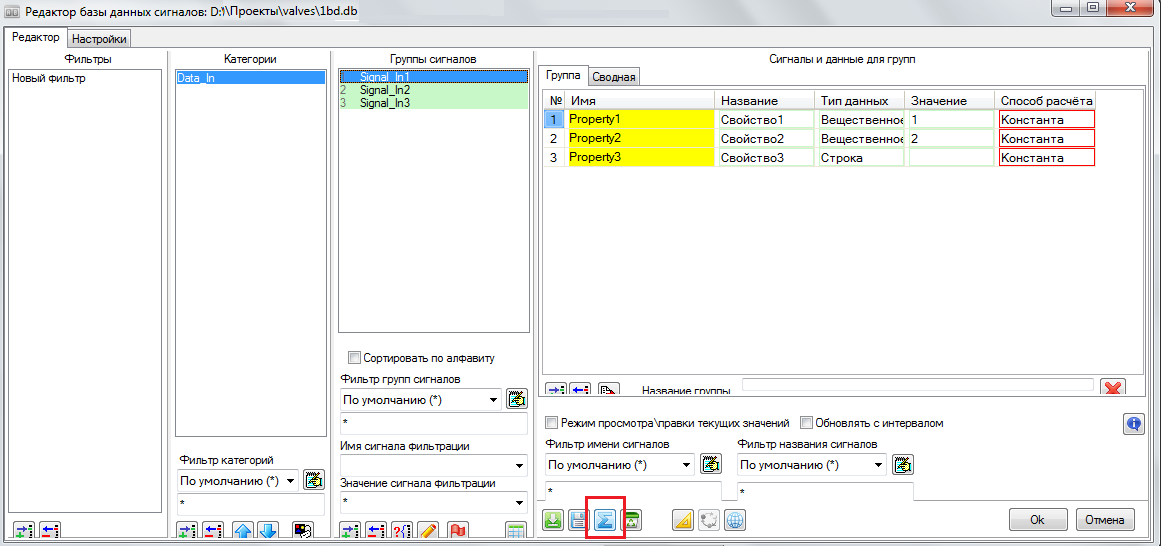
**Рисунок 13 Редактор базы данных пользователя 2 (БД2)**

****

**Рисунок 14 Результат сравнения БД2 и БД1**

Как видно на рисунке 14, категория с измененным шаблоном, все группы сигналов данной категории и добавленное в шаблон новое свойство будут выделены красным цветом в форме сравнения БД2 с БД1. Также красным будет выделена новая группа сигналов. Для продолжения работы необходимо объединить изменения двух пользователей.

Для этого в БД1 удаляем категорию Data\_Out, оставив «правильные» изменения от пользователя 1, а в БД2 удаляем категорию Data\_In (оставляем только категорию Data\_Out). Далее для БД2 вызываем функцию "Дополнить из файла" (см. рисунок 15) и объединяем с БД первого пользователя.

****

**Рисунок 15 Вызов функции "Дополнить из файла"**

В итоге получаем БД с изменениями обоих пользователей - категория Data\_In из БД первого пользователя, а категория Data\_Out из БД второго пользователя.