



## EasyPortal

---

Руководство по эксплуатации  
Версия 1.39

## Оглавление

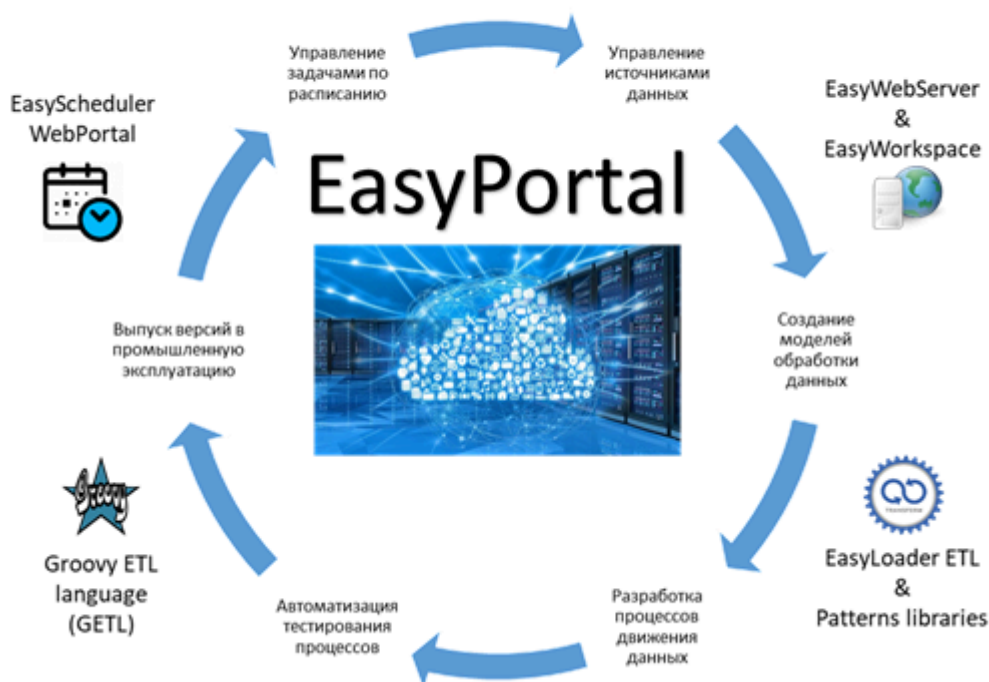
Описание.....	3
Используемые компоненты .....	4
Администрирование.....	5
Требования к системе.....	6
Установка и обновление клиентской части EasyClient для ОС Windows, Linux и MacOS.....	7
Установка и обновление серверной части EasyWebServer для ОС Windows.....	11
Установка и обновление серверной части EasyWebServer для ОС Linux.....	15
Интеграция с EasyScheduler.....	16
Подключение к EasyPortal.....	22
Управление EasyPortal .....	24
Добавление пользователей портала.....	26
Добавление глобальных переменных портала.....	27
Регистрация библиотек шаблонов .....	28
Управление репозиториями .....	29
Примеры работы.....	31
Настройка демонстрационного репозитория DEMO.....	32
Копирование данных из таблиц БД в файлы .....	35
Создание подключения к файловому источнику CSV .....	36
Создание подключения к РСУБД H2.....	39
Создание скриптов инициализации БД H2 .....	41
Создание задачи инициализации БД H2 .....	43
Импорт созданных таблиц из БД H2 в репозиторий .....	47
Создание модели генерации данных в H2 таблицы.....	51
Создание задачи генерации данных в H2 таблицы .....	57
Экспорт структуры H2 таблиц в файловый источник CSV.....	60
Создание модели маппинга данных H2 таблиц и CSV файлов .....	63
Создание задачи копирования данных с таблиц в CSV файлы.....	68
Генерация эталонных данных.....	72
Создание файловых менеджеров эталонных и рабочих CSV файлов.....	73
Создание задачи для генерации архива эталонных CSV файлов .....	76
Создание модели эталонных CSV файлов.....	80

## Описание

С 2013 года компания EasyData развивает линейку российских программных продуктов для автоматизации процессов работы с файловыми системами, РСУБД, хранилищами данными и Big Data платформами. EasyPortal успешно работает в ряде крупных проектов российских компаний в качестве ETL/ELT/Cron платформы и управляет потоками данных десятков терабайт в сутки на хранилищах данных в сотни терабайт. Все продукты зарегистрированы в Российском Реестре Программного Обеспечения.

Платформа EasyPortal покрывает большую часть задач проектов хранилищ данных и сразу предоставляет готовое решение для разработки проектов хранилищ данных и управлением выполнением задач на технической инфраструктуре промышленной среды:

## Экосистема продуктов EasyData



## Используемые компоненты

Все продукты EasyPortal разработаны на open source платформе **Open Java 11** (<https://openjdk.java.net>).

В работе используются следующие библиотеки, фреймворки и компоненты в рамках правил их лицензирования:

- **Apache Groovy**: open source совместимый с Java язык программирования (<https://www.groovy-lang.org>)
- **Spring Framework**: open source фреймворк для Java платформы (<https://spring.io>)
- **GETL**: open source фреймворк разработки ETL процессов для Java платформы (<https://github.com/ascrus/getl>)
- **Webix**: open source фреймворк визуальных компонент Java Script (<https://ru.webix.com>)
- **Quartz Scheduler**: open source библиотека запуска процессов по расписанию (<https://github.com/quartz-scheduler/quartz>)

## Администрирование

Перед началом установки продуктов EasyPortal убедитесь, что Ваша среда соответствует [техническим требованиям и установленная ОС](#) поддерживается продуктами EasyPortal.

## Требования к системе

### Поддерживаемые операционные системы

ОС Windows: 7 SP1/8/8.1/10/11 или Windows Server 2012/2012R2/2016/2019

ОС Linux: Cenos 6.9/7.4/8 или RHEL 6.9/7.4/8 или SLES 12 или Ubuntu 16.04/18.04/20.04

ОС macOS (\*): OSX 10.10+ (x64)

\* Только для инсталляции EasyClient

### Требования к аппаратной части

Продукт	Процессор	Доступная память (*)	Свободное дисковое пространство
EasyPortal WebServer	от 4 ядер	от 4 гб	от 10 гб
EasyPortal Client	от 2 ядер	от 2 гб	от 5 гб
EasyScheduler	от 4 ядер	от 2 гб	от 10 гб
EasyRuntime	от 4 ядер	от 1 гб	от 10 гб

\* Не учитывая память, которая потребуется для отладки пользователями процессов workflow репозитория EasyPortal или программ ОС при их вызове с EasyRuntime.

### Требования к установленному ПО

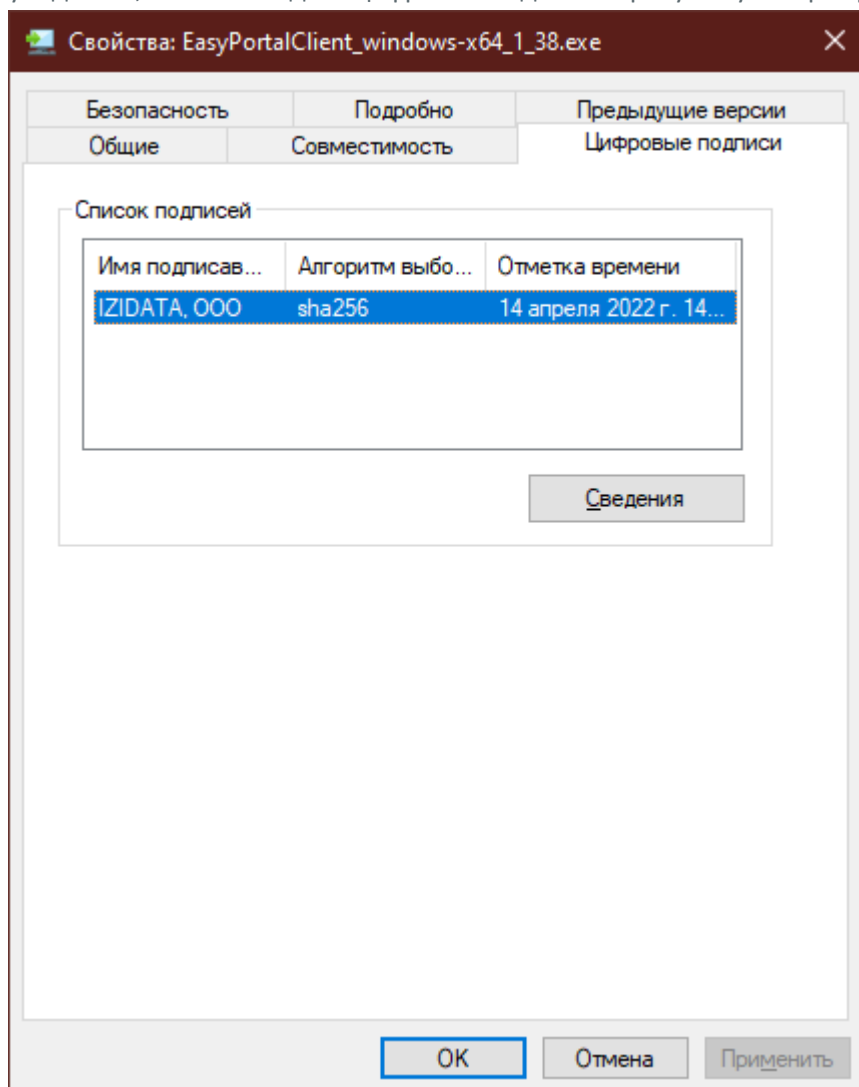
Для запуска рабочего места на Linux должен быть установлен и настроен X-Server.

### Требования к Firewall

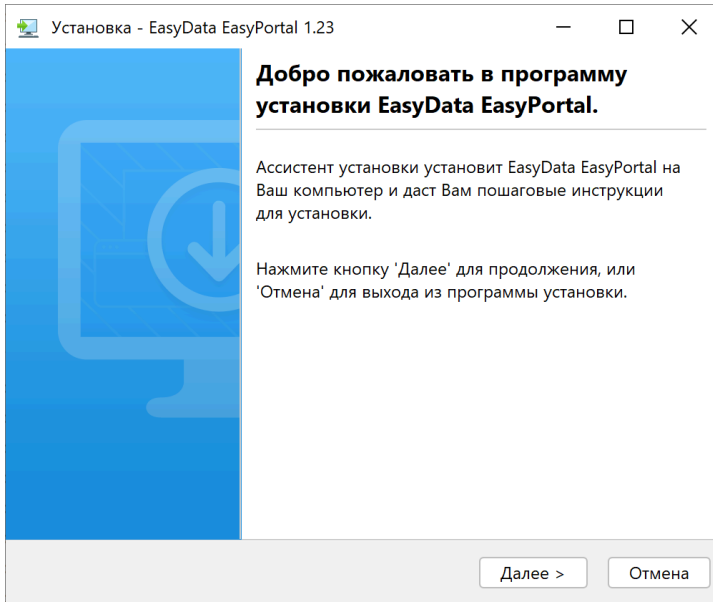
Для удаленной работы пользовательских машин с сервером EasyWebServer должен быть открыт порт 9050.

## Установка и обновление клиентской части EasyClient для ОС Windows, Linux и MacOS

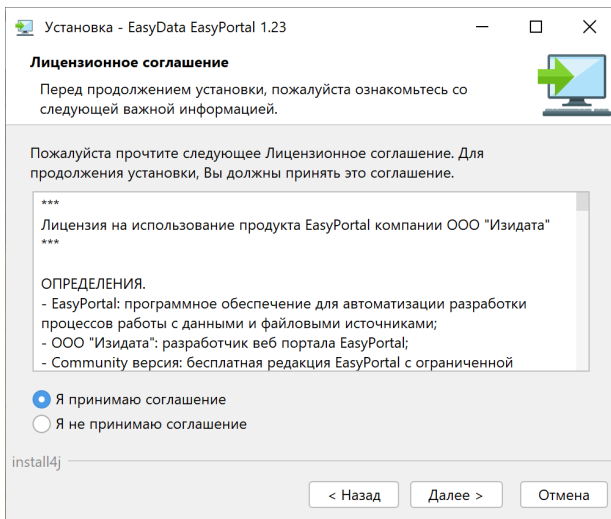
Скачайте последнюю версию EasyClient для нужной ОС с сайта [easyportal.ru](http://easyportal.ru) и запустите инсталлятор. В случае предупреждения Windows Defender о безопасности установки выберите "Дополнительно" и "Разрешить установку". Все программное обеспечение компании EasyData зарегистрировано в центре сертификации ПО и гарантирует безопасное выполнение. Для проверки сертификата откройте в проводнике Windows свойства файла инсталляции и убедитесь, что во вкладке "Цифровые подписи" присутствует сертификат "IZIDATA":



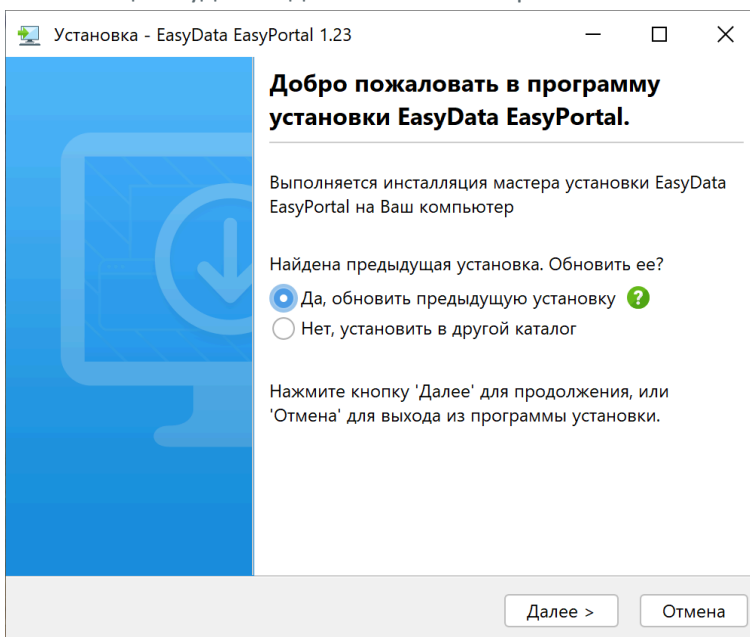
Нажмите кнопку **Далее** на первом экране:



Прочитайте и примите лицензионное соглашение:

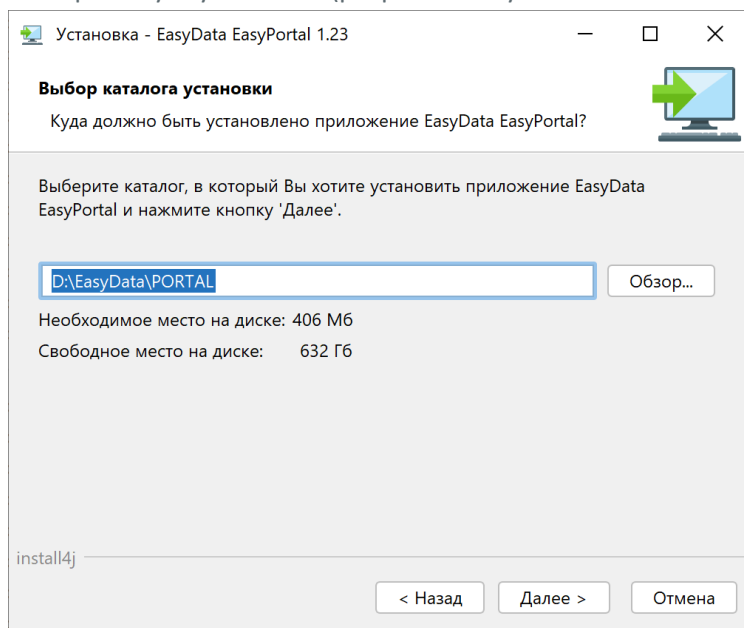


Если ранее программа была установлена, то появится экран запроса обновление установки. При выборе новой инсталляции будет создана новая копия рабочего места:

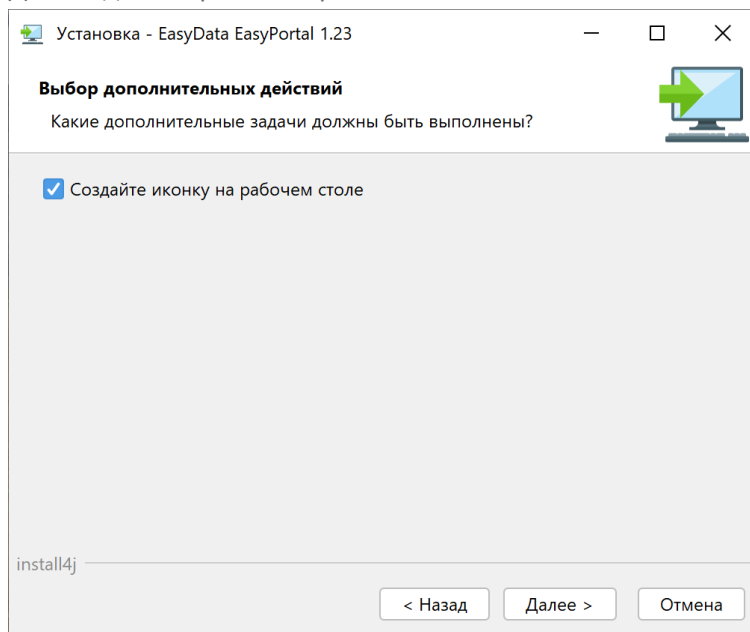




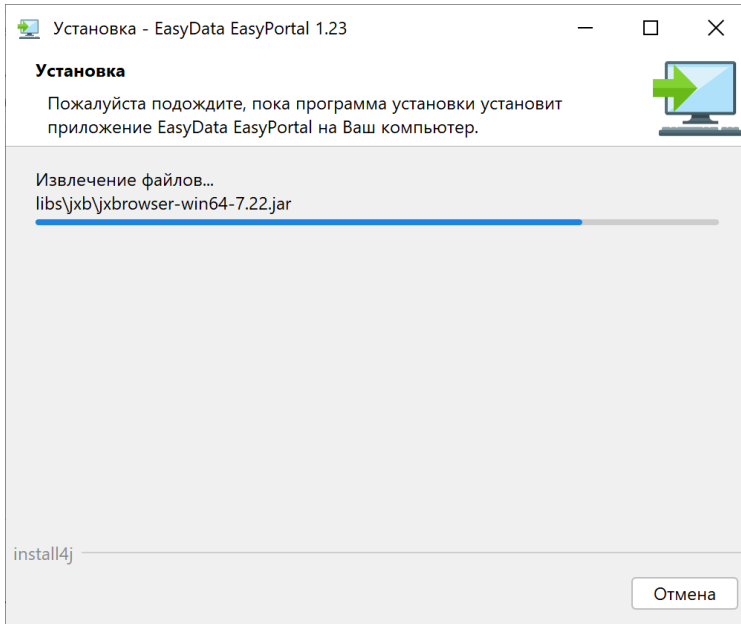
Выберите путь установки (разрешается установка в пользовательский каталог без прав администратора):



Для создания ярлыка на рабочем столе пользователя поставьте галочку:



Дождитесь окончания установки:



После установки в командной строке появятся программы:

- **В командной строке:**

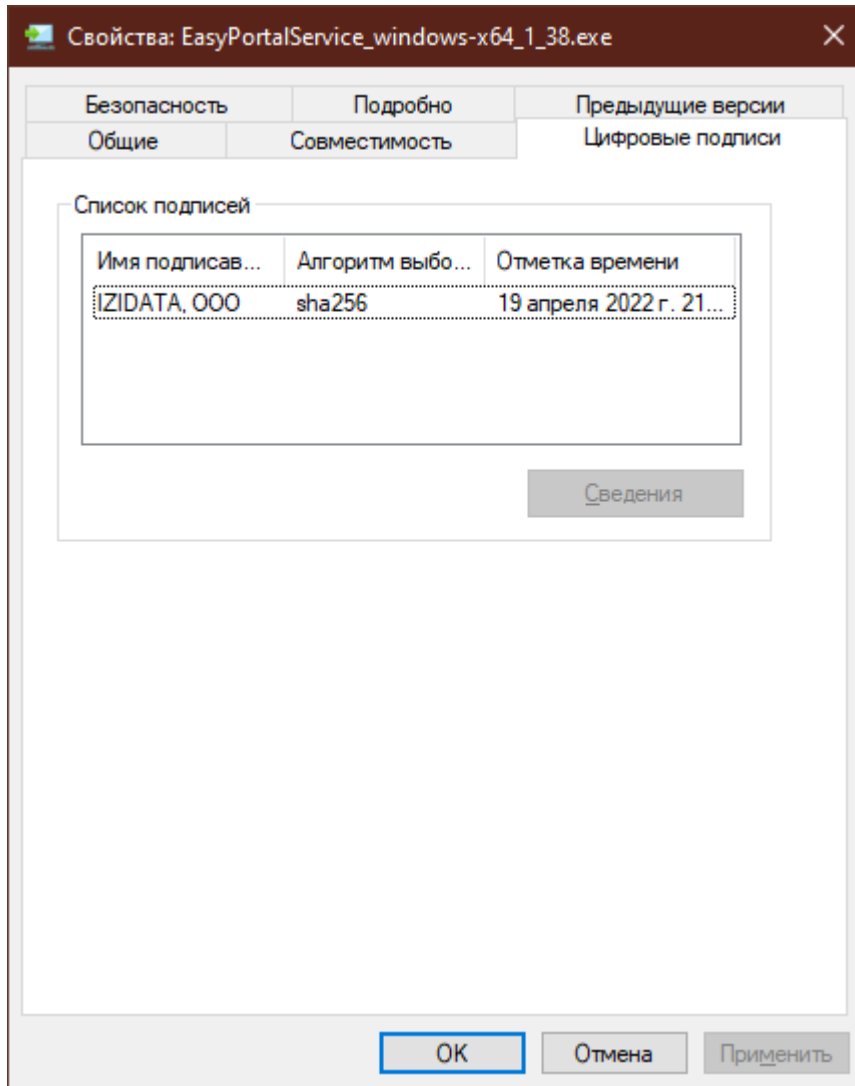
- EasyLoader - консольная программа для запуска задач репозитория EasyPortal;
- GroovyConsole - визуальная консоль Groovy для интерактивной работы с объектами репозитория EasyPortal;

- **В меню Пуск группы EasyData EasyPortal:**

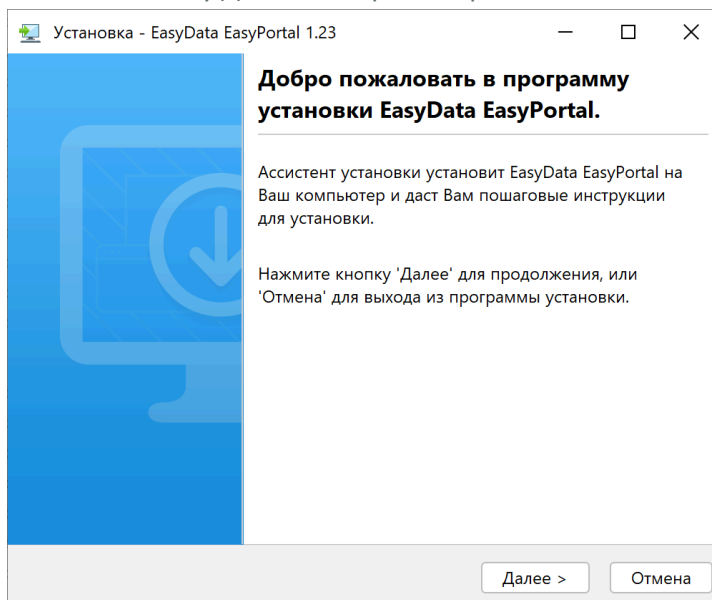
- a. EasyWorkspace - рабочее место для запуска и работы с локальными репозиториями EasyPortal и с центром управления задачами EasyScheduler;
- GroovyConsole - визуальная консоль Groovy для интерактивной работы с объектами репозитория EasyPortal.

## Установка и обновление серверной части EasyWebServer для ОС Windows

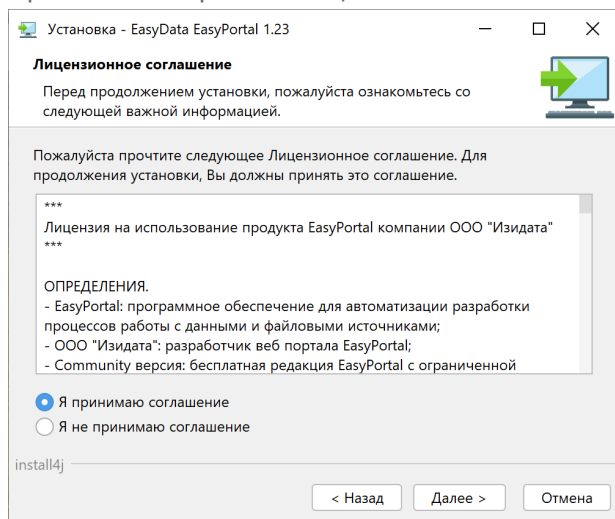
Скачайте последнюю версию EasyWebServer для Windows с сайта [easyportal.ru](http://easyportal.ru) и запустите инсталлятор. В случае предупреждения Windows Defender о безопасности установки выберите "Дополнительно" и "Разрешить установку". Все программное обеспечение компании EasyData зарегистрировано в центре сертификации ПО и гарантирует безопасное выполнение. Для проверки сертификата откройте в проводнике Windows свойства файла инсталляции и убедитесь, что во вкладке "Цифровые подписи" присутствует сертификат "IZIDATA":



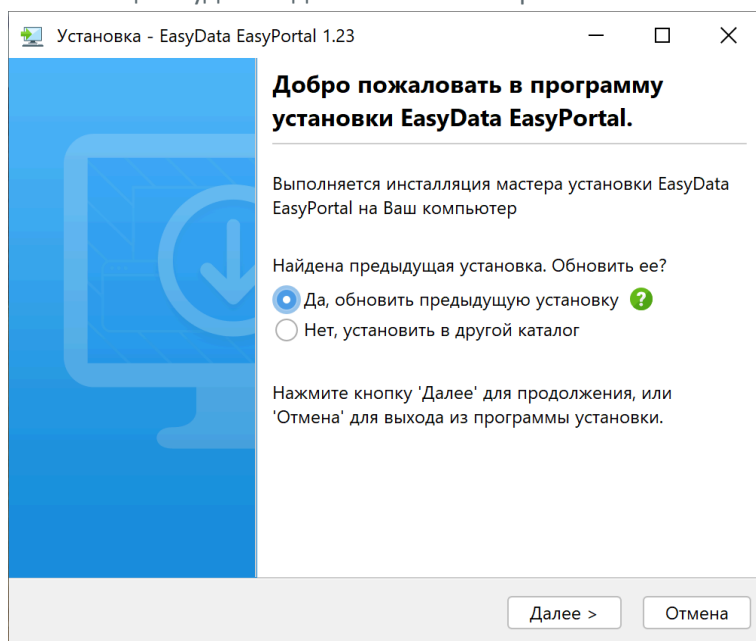
Нажмите кнопку Далее на первом экране:



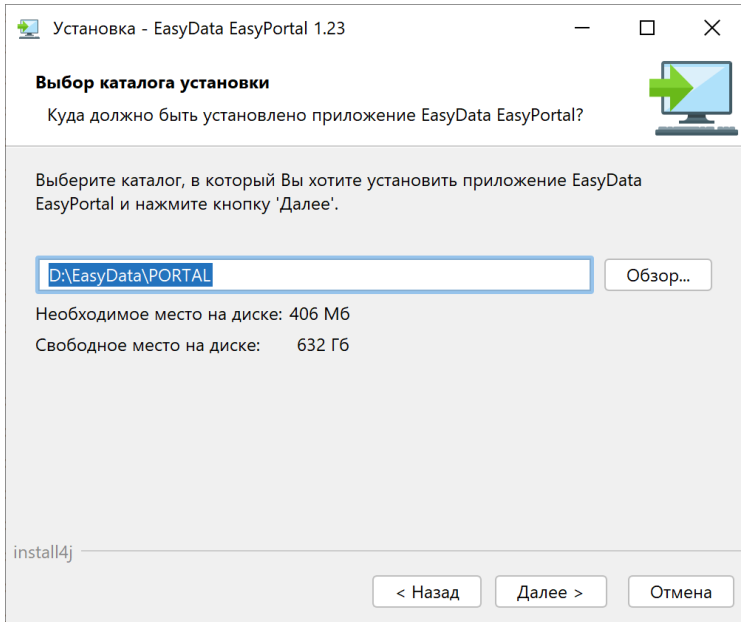
Прочитайте и примите лицензионное соглашение:



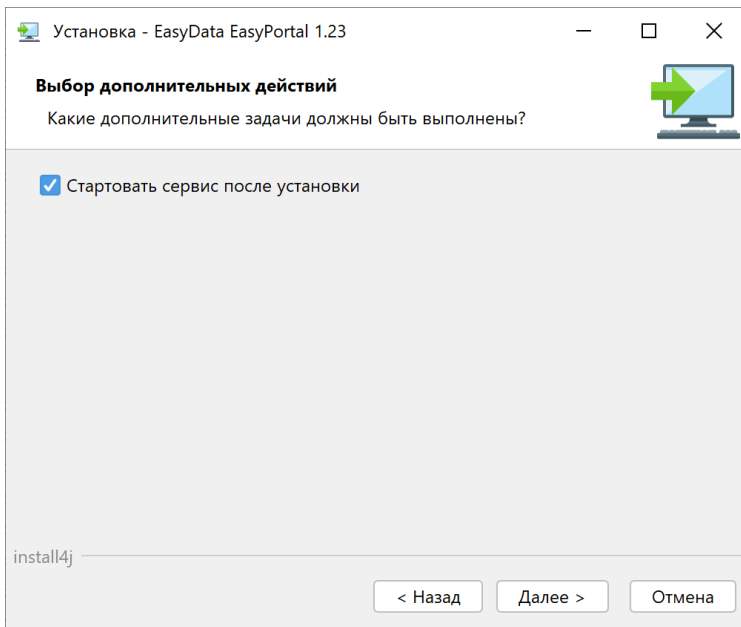
Если ранее программа была установлена, то появится экран запроса обновление установки. При выборе новой инсталляции будет создана новая копия рабочего места:



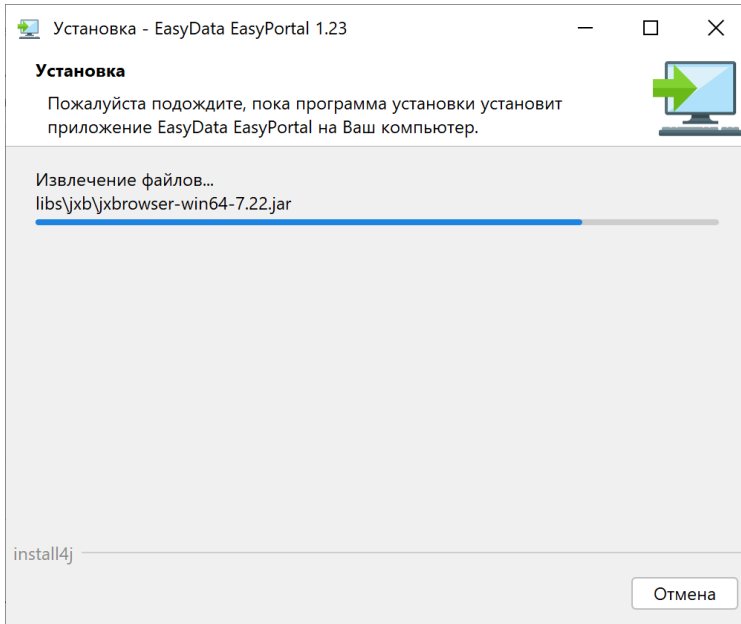
Выберите путь установки (установка требует права администратора):



Для автоматического старта сервиса Windows после установки EasyPortal поставьте галочку:



Дождитесь окончания установки:



После установки в командной строке появятся программы:

- **В командной строке:**
  - EasyLoader - консольная программа для запуска задач репозитория EasyPortal;
- **В меню Пуск группы EasyData EasyPortal:**
  - а. EasyWorkspace - рабочее место для запуска и работы с локальными репозиториями EasyPortal и с центром управления задачами EasyScheduler;
  - GroovyConsole - визуальная консоль Groovy для интерактивной работы с объектами репозитория EasyPortal.

Проверьте работоспособность сервера после инсталляции и запуска сервиса. Для этого в браузере откройте ссылку: <https://localhost:9050>



## Установка и обновление серверной части EasyWebServer для ОС Linux

Скачайте последнюю версию EasyWebServer для Linux с сайта [easyportal.ru](http://easyportal.ru), скопируйте на сервер установочный пакет.

### RedHat/CentOS

#### Установка

Для новой установки службы `easyportal.service` в командной строке запустите следующую команду:

```
sudo dnf install /Путь/EasyPortalService_linux_1_38.rpm
```

где "Путь" это путь до пакета инсталляции. В ходе инсталляции нажмите Y.

#### Обновление

Для обновления версии службы `easyportal.service` в командной строке запустите следующую команду:

```
sudo dnf update /Путь/EasyPortalService_linux_1_38.rpm
```

где "Путь" это путь до пакета инсталляции. В ходе инсталляции нажмите Y.

### Debian

#### Установка

Для новой установки службы `easyportal.service` в командной строке запустите следующую команду:

```
sudo apt install /Путь/EasyPortalService_linux_1_38.deb
```

где "Путь" это путь до пакета инсталляции. В ходе инсталляции нажмите Y.

#### Обновление

Для обновления версии службы `easyportal.service` в командной строке запустите следующую команду:

```
sudo apt upgrade /Путь/EasyPortalService_linux_1_38.deb
```

где "Путь" это путь до пакета инсталляции. В ходе инсталляции нажмите Y.

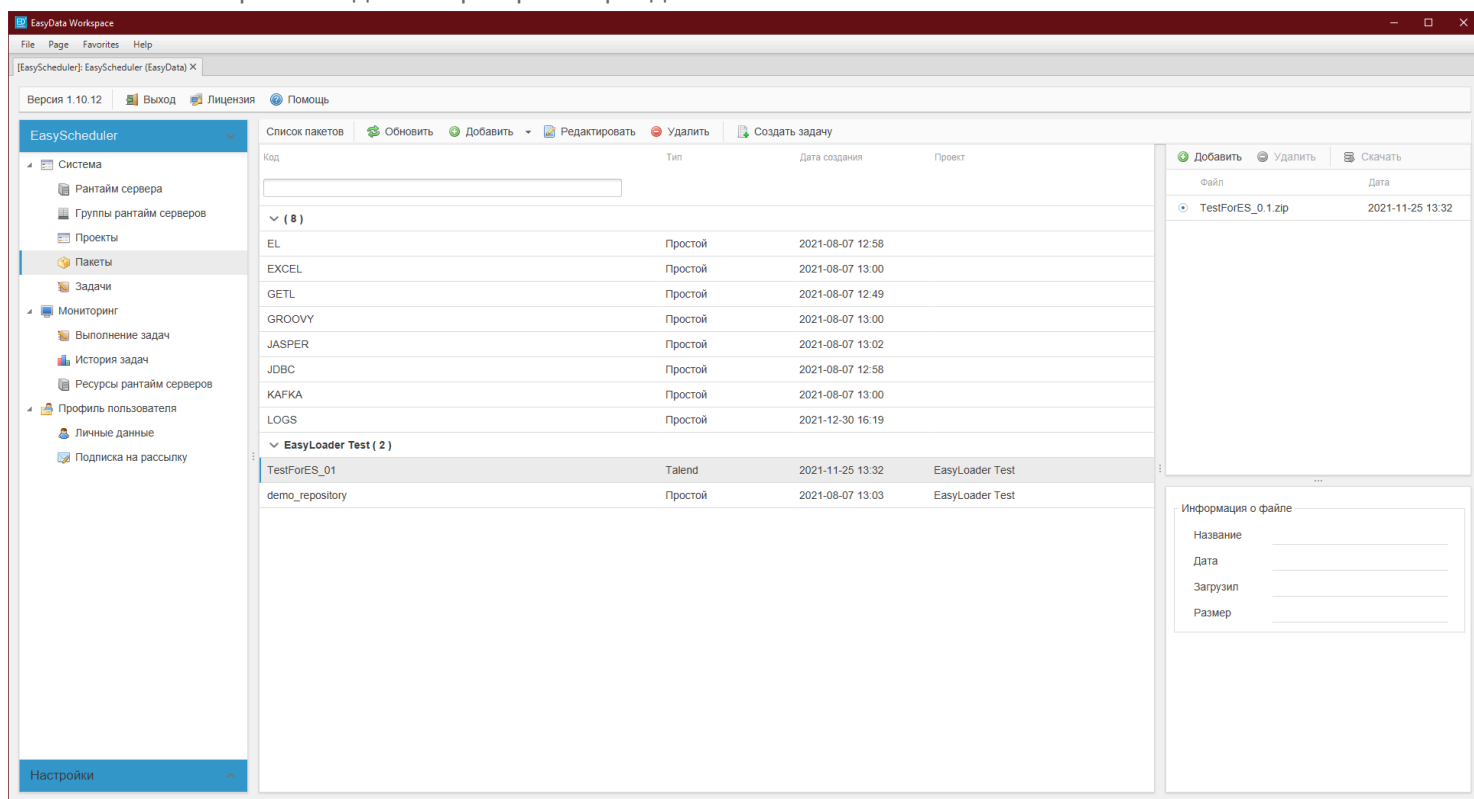
## Интеграция с EasyScheduler

Для организации запуска задач по расписанию и контролем за их выполнением EasyPortal включает в себя центр управления задачами EasyScheduler. Продукт позволяет подключить все эксплуатирующиеся сервера в единый интерфейс и управлять выполнением на них задач:



## Установка пакета EasyLoader



Для запуска задач, разработанных в репозитории EasyPortal требуется установить инсталляцию runtime компонента выполнения задач EasyLoader как пакет EasyScheduler. Для этого скачайте пакеты EasyLoader для нужной платформы ОС с сайта <https://www.easyportal.ru>. Подключитесь к EasyScheduler в браузере или через EasyWorkspace под пользователем с правами администратора и перейдите в пакеты:



Добавьте новый простой пакет с названием "EL\_WIN":



Создать простой пакет ×

 Выход |  Сохранить

Код \*   
Допускаются только английские буквы, цифры и символы - \_

Проект

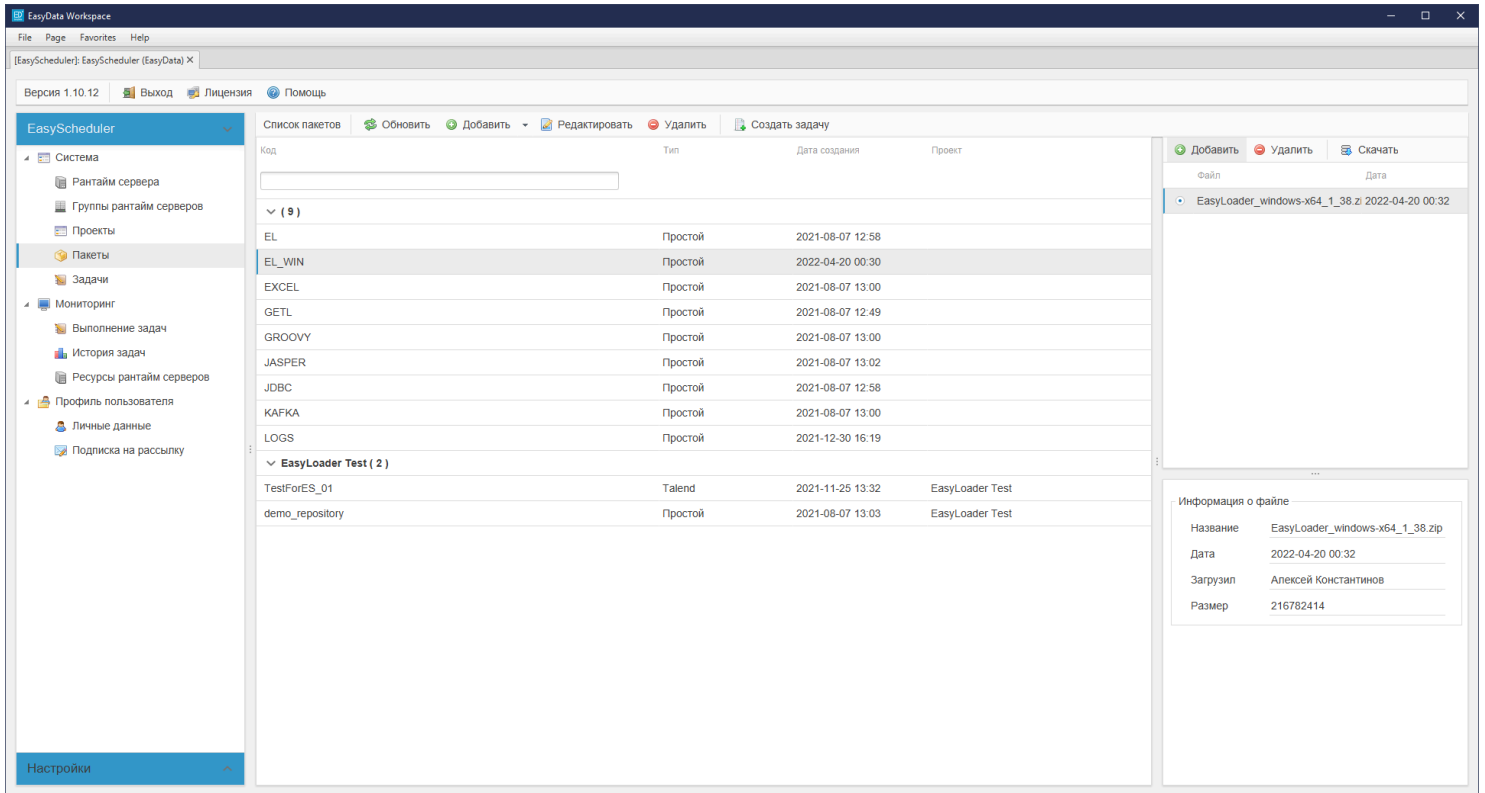
**Описание**

**B I U X**

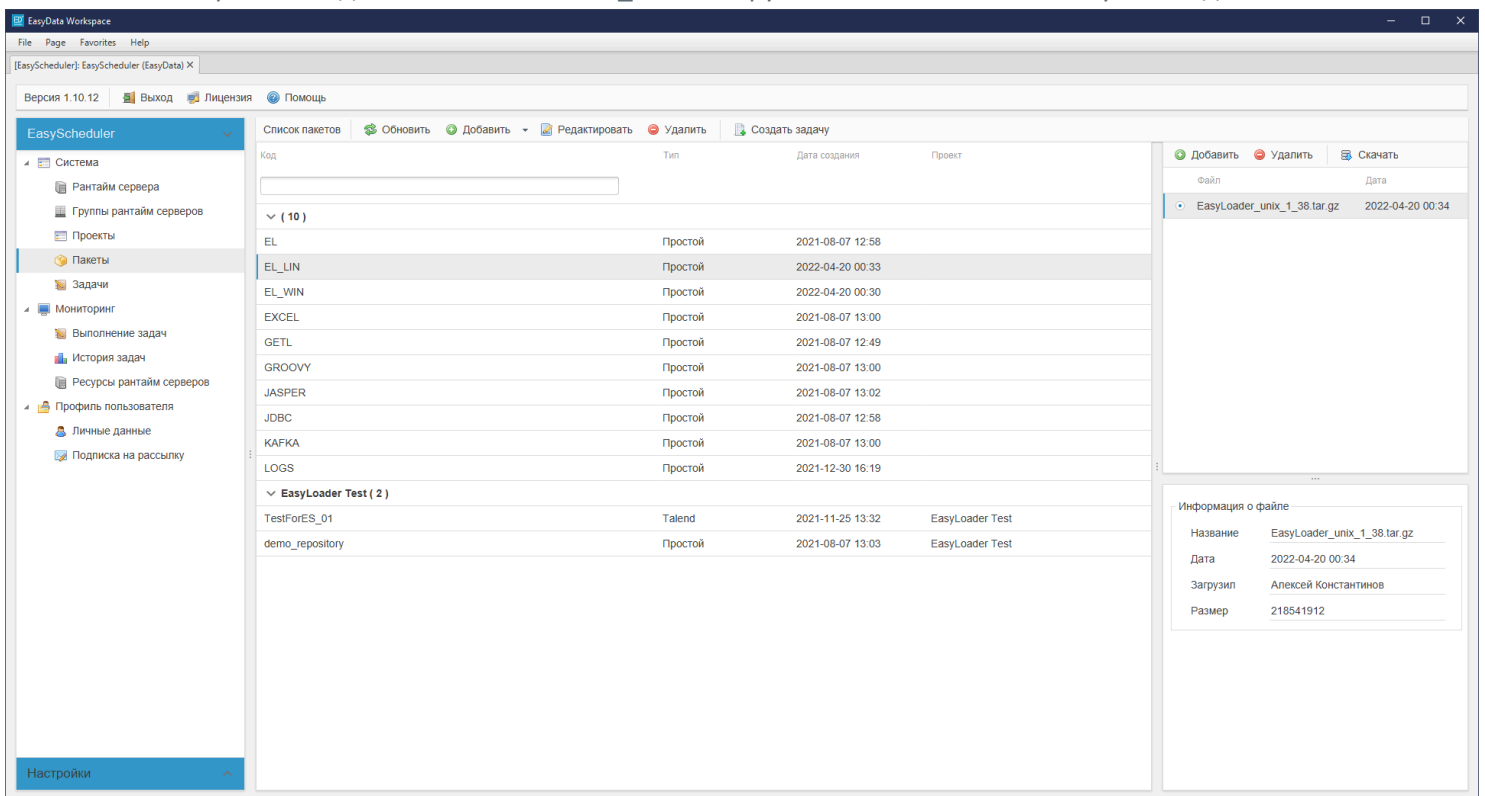
EasyLoader для Windows

P.S. Можно задать любое имя пакету, в данном случае здесь в качестве имени указано сокращенное EasyLoader for Windows.

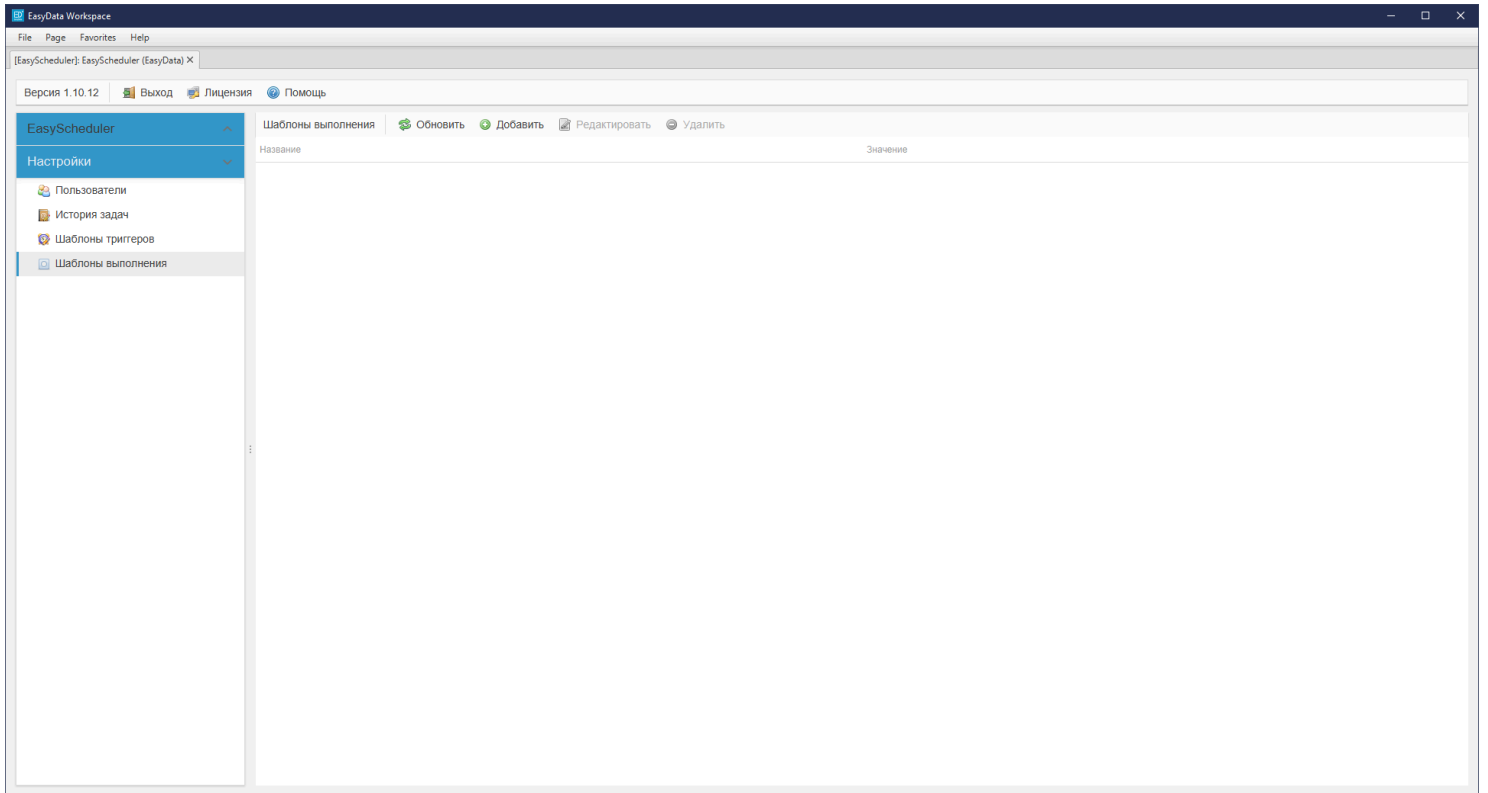
Загрузите в пакет скаченный EasyLoader для ОС Windows:



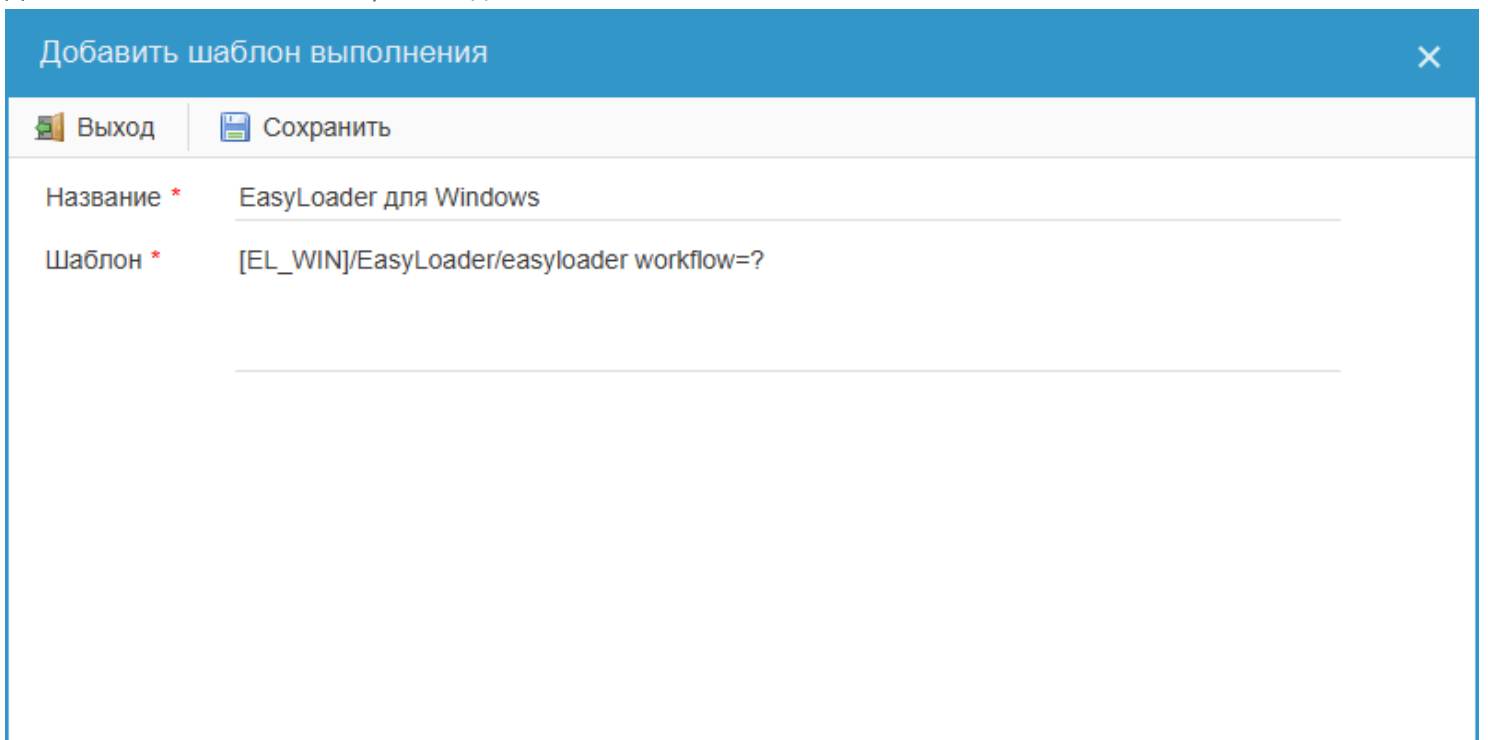
Аналогичным образом создайте новый пакет "EL\_LIN" и загрузите в него скаченный EasyLoader для Linux:



Переключитесь в "Настройки" и выберите там "Шаблоны":



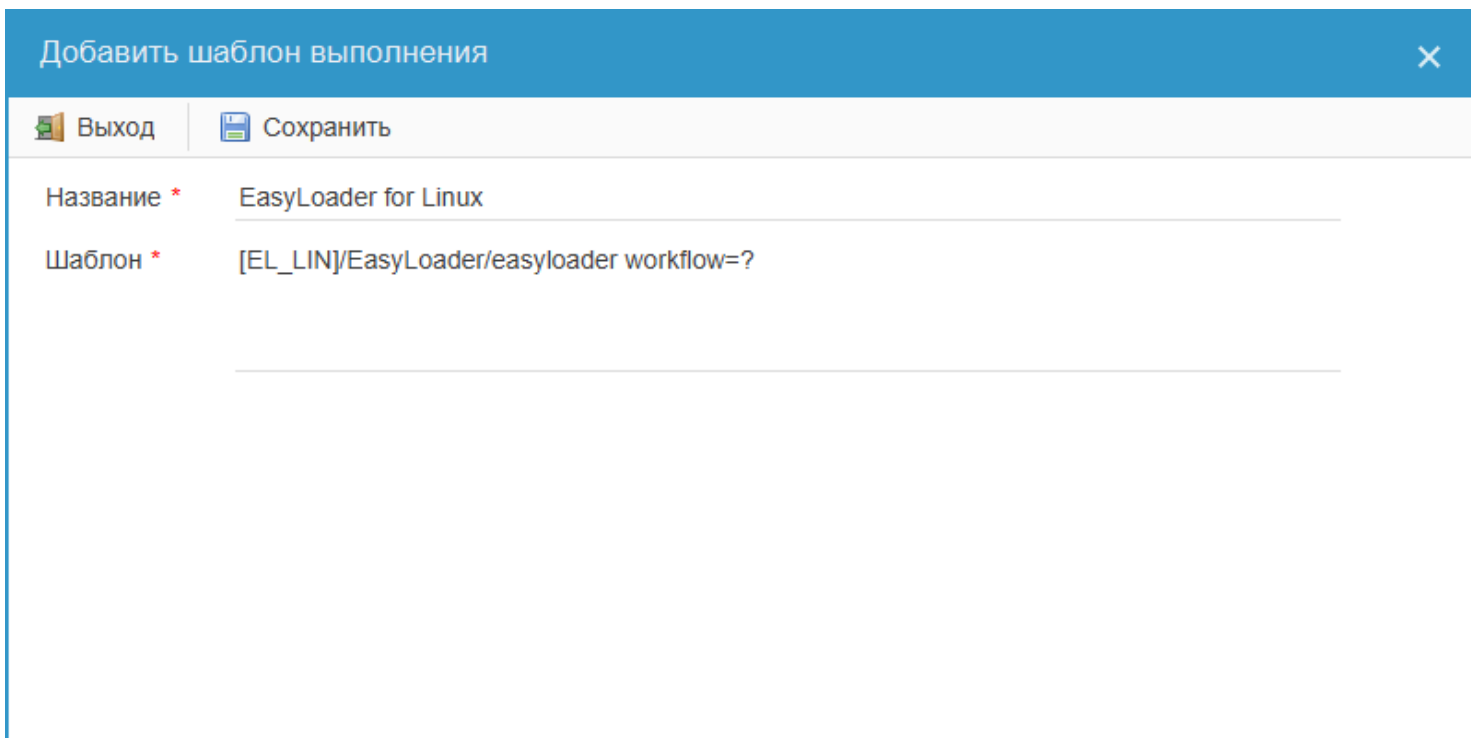
Добавьте новый шаблон "EasyLoader для Windows":



В строку выполнения шаблона вбейте:

`[EL_WIN]/EasyLoader/easyloader workflow=?`

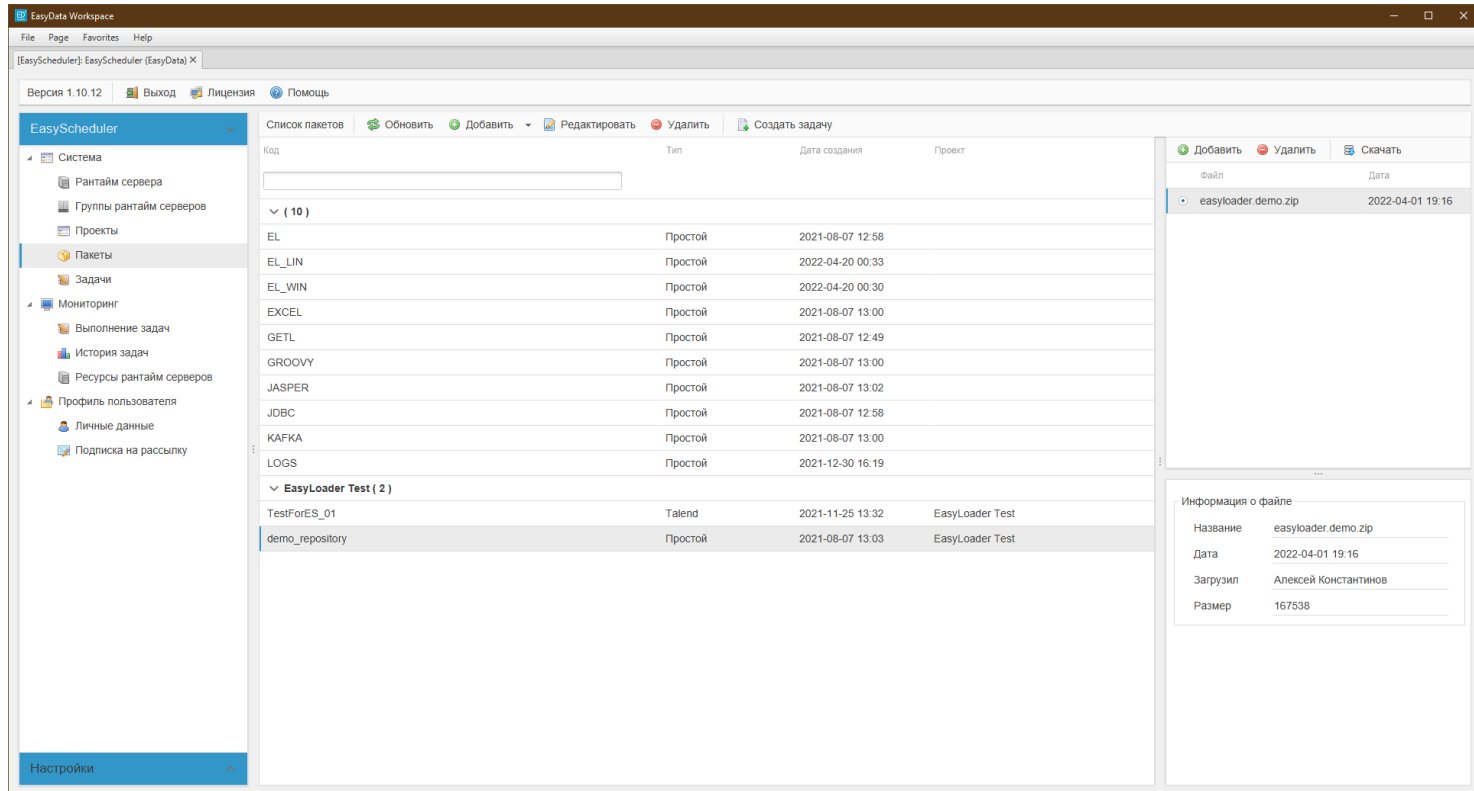
Аналогично добавьте новый шаблон "EasyLoader for Linux":



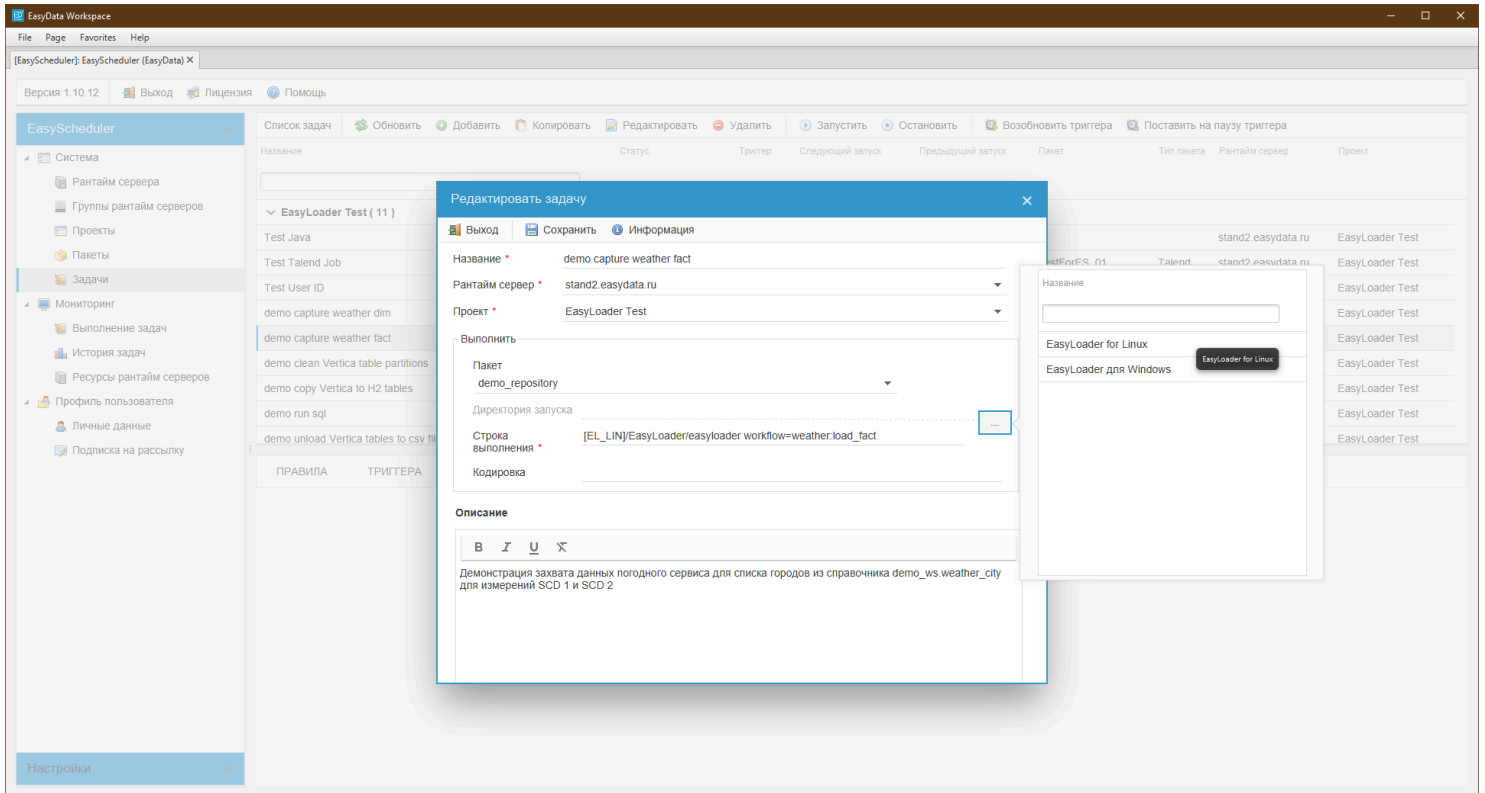
В строку выполнения шаблона вбейте:  
`[EL_LIN]/EasyLoader/easyloader workflow=?`

## Запуск задач с репозитория EasyPortal

Чтобы запускать задачи потребуется загрузить объекты репозитория EasyPortal в EasyScheduler. Для этого в панели администратора EasyPortal выгрузите репозиторий в zip файл и загрузите его как пакет EasyScheduler в нужный проект:



Теперь можно добавлять на выполнение в EasyScheduler задачи EasyPortal:



Для этого потребуется при создании новой задачи:

1. Выбрать в пакете задачи загруженный пакет репозитория;
2. Нажать на кнопку шаблонов и выбрать шаблон, который соответствует ОС рантайм сервера;
3. Вместо "workflow=?" вписать "workflow=имя задачи в репозитории".

После создания Вы можете добавить задачи триггера и правила, запустив задачу в автоматическое выполнение по заданному расписанию.

## Обновление версии EasyLoader

Для обновления EasyLoader скачайте новую версию под нужную ОС с сайта <https://www.easyportal.ru> и добавьте новую версию в нужный пакет (EL\_WIN или EL\_LIN).

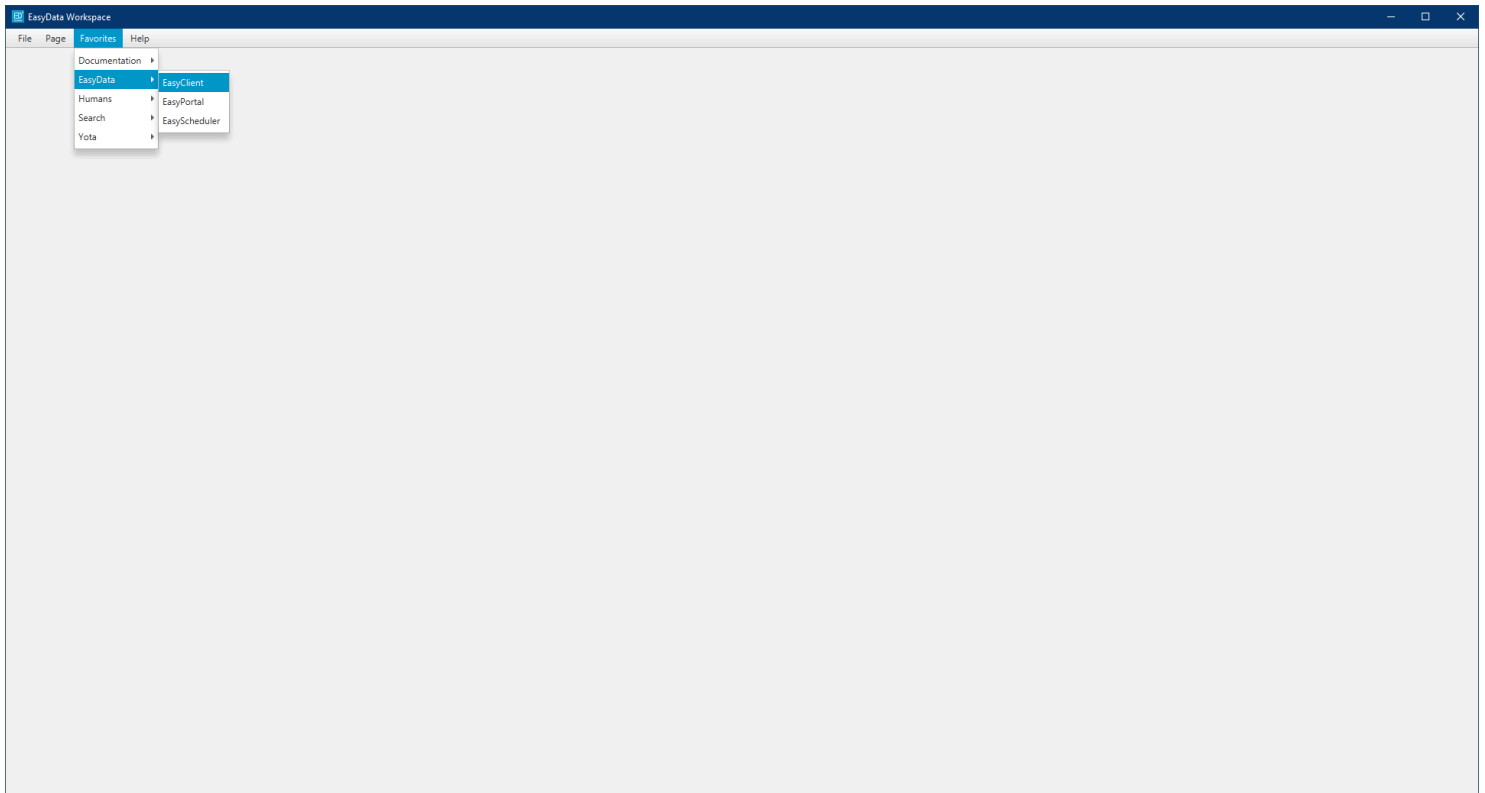
## Обновлении репозитория EasyPortal

Для обновления репозитория с задачами EasyPortal в панели администратора EasyPortal скачайте новый пакет репозитория и загрузите полученный zip файл в пакет репозитория EasyScheduler.

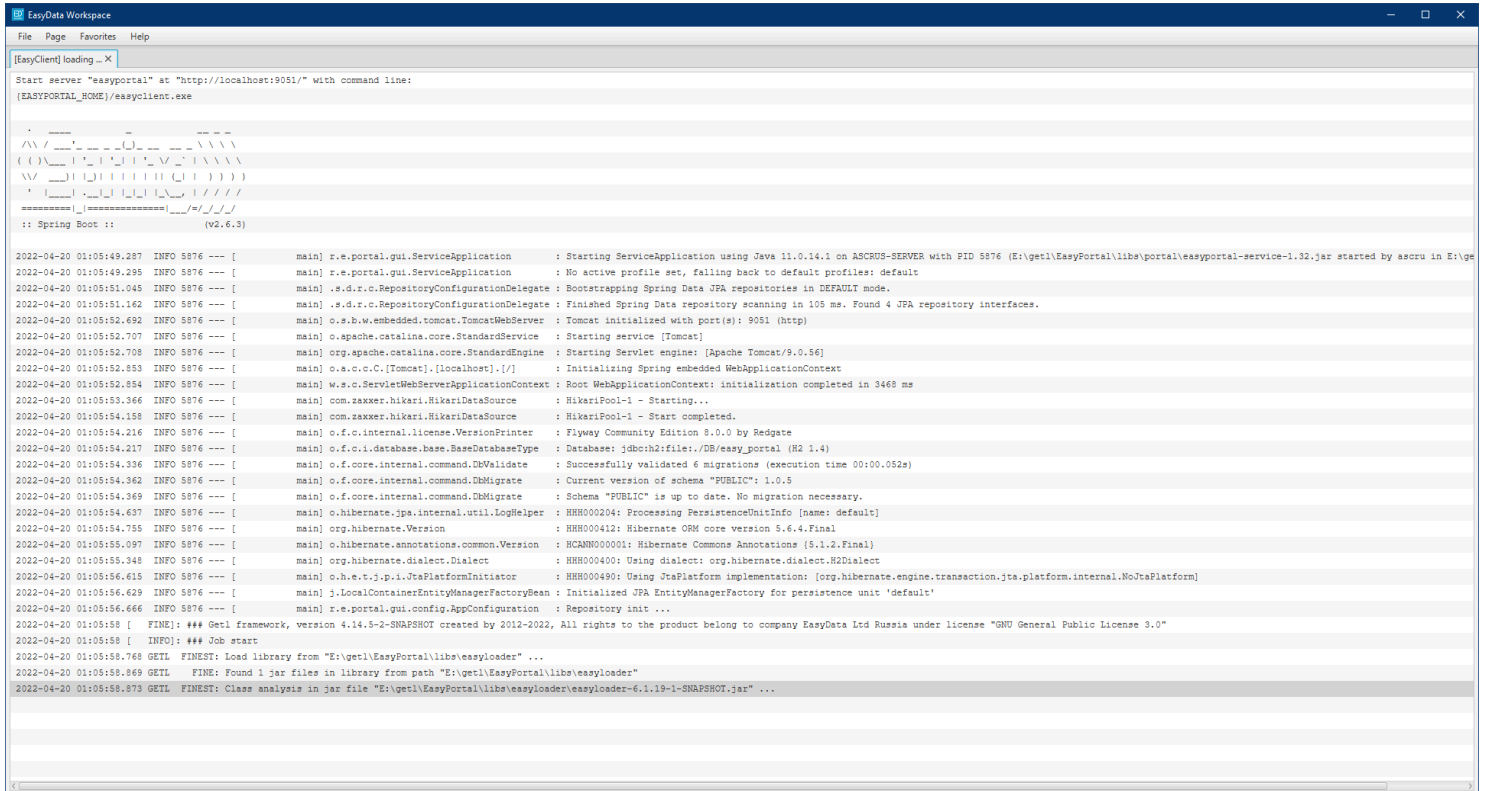
## Подключение к EasyPortal

### Работа в локальном сервером с помощью EasyWorkspace

Для подключения к установленной локальной версии сервера запустите EasyWorkspace, в меню "Favorites" выберите пункт "EasyData" и подпункт "EasyClient":



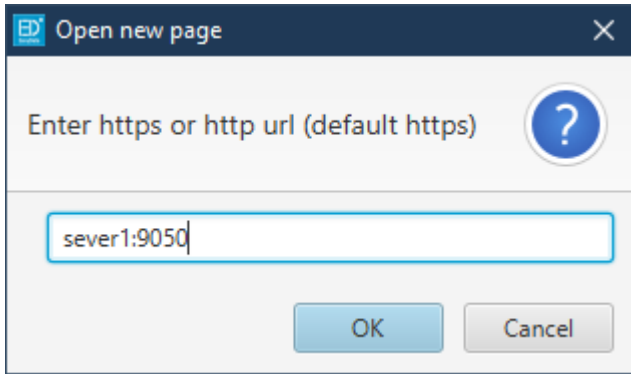
Автоматически стартует локальный сервис, где будет показываться лог его работы до полного запуска:



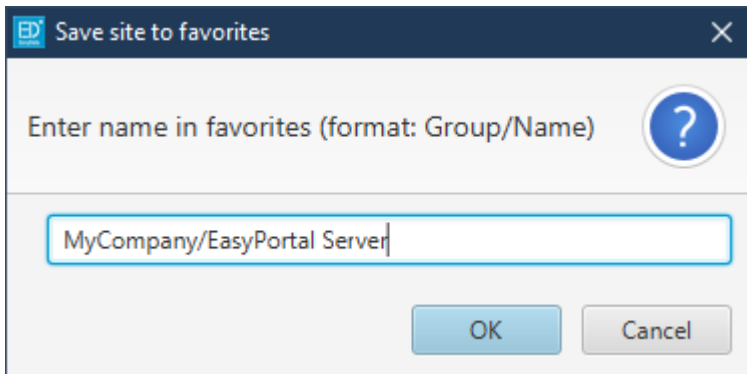
После полной загрузки интерфейс переключится на локальный веб интерфейс и выдет приглашение регистрации.

### Работа с удаленным сервером EasyWebServer с помощью EasyWorkspace

Для подключения к EasyWebServer запустите EasyWorkspace, в меню "File" выберите пункт "Open" и введите URL подключения к серверу:



Где sever1 адрес хоста установленного сервера. При нажатии "OK" откроется интерфейс доступа к серверу. Для сохранения закладки к серверу в фаворитах в меню "File" выберите пункт "Save to favorites" и в открывшемся окне введите "ИмяГруппы/Имя вкладки":



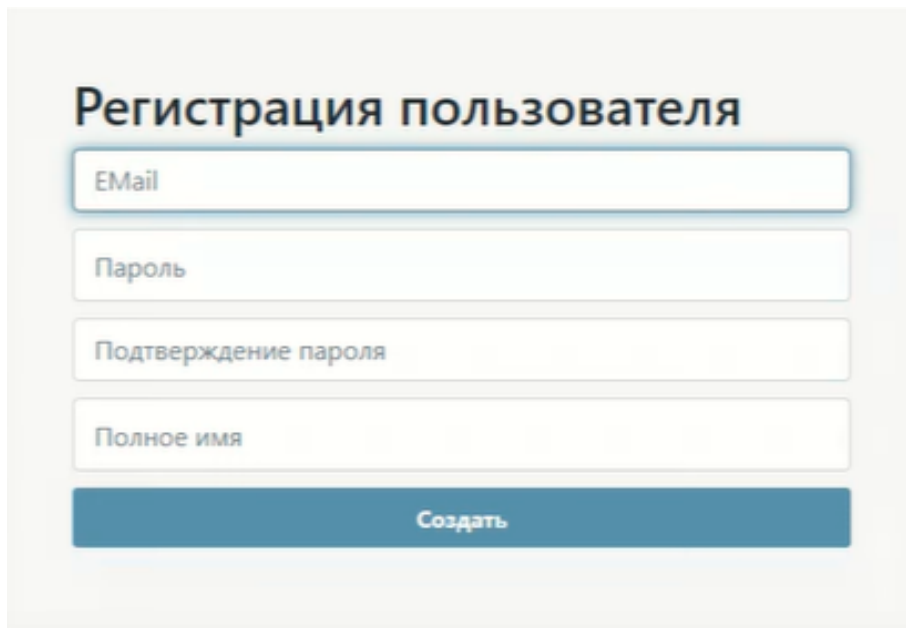
P.S. Имя группы вводится обязательно и разделяется от имени сохраненной вкладки слэшем.

## Работа с удаленным сервером EasyWebServer с помощью браузеров

В своем браузере введите URL подключения к серверу "https://sever1:9050", где sever1 - хост сервера EasyWebServer. После появления окна интерфейса сохраните вкладку в Избранное.

## Управление EasyPortal

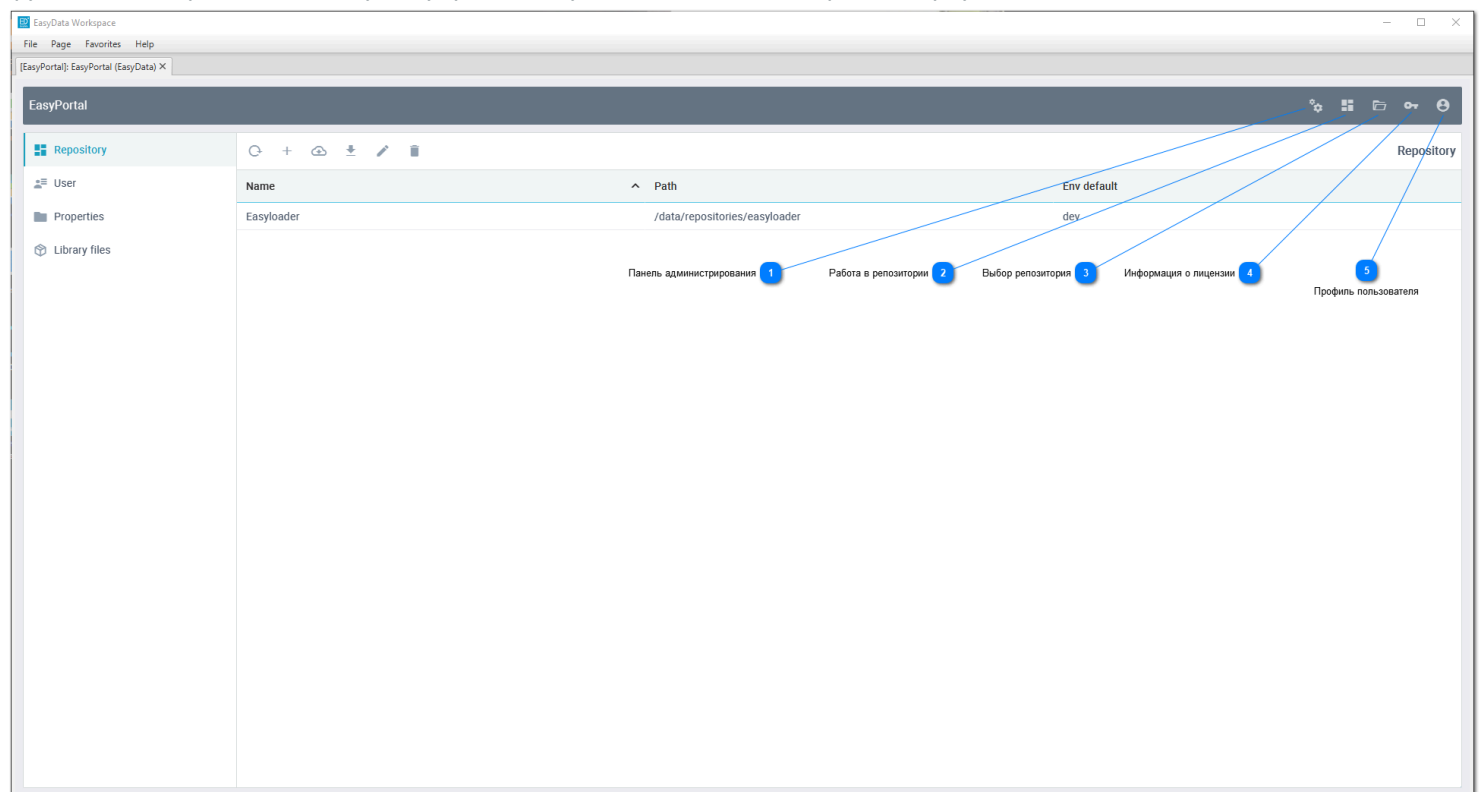
После первичной инсталляции EasyPortal при подключении к нему появится окно регистрации его администратора. Заполните все поля и нажмите "Создать":



The screenshot shows a registration window titled "Регистрация пользователя" (User Registration). It contains four input fields: "EMail", "Пароль" (Password), "Подтверждение пароля" (Confirm password), and "Полное имя" (Full name). Below the fields is a blue button labeled "Создать" (Create).

P.S. Указывайте работающий почтовый ящик, чтобы можно было получать уведомления от портала и восстановить при необходимости доступ к системе.

После регистрации и успешной авторизации открывается рабочее пространство EasyPortal. Для управления функциями портала используйте управляющую панель кнопок справа сверху:



### 1 Панель администрирования



Управление репозиториями, пользователями, глобальными переменными и библиотеками шаблонов.



## 2 Работа в репозитории



Работа с объектами выбранного репозитория.

## 3 Выбор репозитория



Выбрать для работы репозиторий.

## 4 Информация о лицензии



Просмотреть текущую лицензию репозитория.

## 5 Профиль пользователя



Изменить данные профиля или выйти с него.

## Добавление пользователей портала

Перейдите в панель администрирования, выберите список пользователей и нажмите кнопку создания (+):

EasyPortal

Repository

User

EMail

Создать

Введите описание нового пользователя:

Create user

EMail	<input type="text"/>	ROLES	
Full Name	<input type="text"/>	Admin	<input type="checkbox"/>
Password	<input type="password"/>	Api	<input type="checkbox"/>
Password Repeated	<input type="password"/>		
Description	<p><u>Подчеркивание</u>   <b>Жирный</b>   <i>Курсив</i></p> <div style="border: 1px solid #ccc; height: 150px;"></div>		
<input type="button" value="Сохранить"/>		<input type="button" value="Закреть"/>	

## Добавление глобальных переменных портала

Чтобы жестко не привязывать работу задач к абсолютному пути файловой системы рекомендуется использовать переменные ОС (environments). В таком случае разрабатываемая задача будет использовать в файловом пути {Переменную}, в которую будет подставляться текущее значение переменной ОС. Это позволит запускать разработанную задачу на любой машине, вне зависимости от среды: от разработки и тестирования до промышленной эксплуатации и без привязки к ОС. Если по каким то причинам в текущей инсталляции EasyPortal не удастся создать переменную среды ОС (например, не хватает прав администратора), но эта переменная должна быть использована в репозитории, можно эмулировать её присутствие с помощью глобальных переменных EasyPortal.

Для добавления глобальной переменной перейдите в панель администрирования и нажмите кнопку создания (+):

**Create property**

<b>Name</b>	<input type="text" value="MyFilePath"/>
<b>Value</b>	<input type="text" value="/data/files"/>

P.S. Имена переменных являются регистрозависимыми. Для локальной инсталляции EasyClient можно также создавать пользовательские переменные ОС для Windows и в Linux.

## Регистрация библиотек шаблонов

EasyPortal включает в себя библиотеку шаблонов EasyLoader, в которую входит множество скриптов с алгоритмами работы с файловыми системами, обработкой данных РСУБД и расширенной работой с хранилищами данных на базе платформа Micro Focus Vertica. Если в рамках задач проекта требуется расширить функциональность дополнительными шаблонами, то после их разработки и сборки в jar файлы, они упаковываются в zip архив и могут быть подключены для работы в портал.

Для подключения разработанной библиотеки шаблонов перейдите в панель администрирования, выберите библиотеку шаблонов и нажмите кнопку создания (+). После выбора zip файла библиотека с прописанным в её конфигурации именем будет подключена к portalу и доступна для работы с её скриптами в репозиториях.

## Управление репозиториями

Вся работа с источниками данных и задачами описывается в репозитории EasyPortal. Репозиторий представляет из себя файловое хранилище описания объектов (метаданных) по указанному пути. Одна инсталляция EasyPortal может иметь более одного репозитория, между которыми можно переключаться при работе. В одной директории может храниться только один репозиторий. Если требуется командная работа с репозиторием различных пользователей с их машин, то репозиторий регистрируется для хранения в системе контроля версий Git, где пользователи могут получать и сохранять его изменения. В ходе разработки можно создавать новые ветки в Git, чтобы разделить разработку новых задач и сопровождение существующих в проекте. Таким образом, даже в локальной инсталляции портала EasyClient может быть много репозитория, как для разных проектов, так и для одного проекта, но его разных версий.

### Создание нового репозитория

Для добавления нового репозитория перейдите в панель администрирования, выберите репозитории и нажмите кнопку создания (+):

#### Create repository

Name	<input type="text" value="MyRepDev"/>
Path	<input type="text" value="/data/repositories/myrep-dev"/>
Password	<input type="text" value="repository password"/>
Env default	<input type="text" value="dev"/>

#### Envs

	Value	
	dev	
	test	
	prod	

Сохранить Закреть

#### Свойства репозитория:

- **Имя** - имя, под которым репозиторий будет зарегистрирован в портале. Если используется Git, то к имени можно дописать имя ветки Git, чтобы в дальнейшем не путаться в именах, если будет подключено несколько разных каталогов разных веток одного Git проекта;
- **Путь** - полный путь хранения в файловой системе файлов репозитория. Если каталог не существует, он будет создан. Если в каталоге уже есть файлы другого репозитория, будет выдана ошибка.
- **Пароль** - пароль кодирования персональной информации, хранящейся в репозитории. Длина пароля должна быть не менее 16 и не более 32 букв. Обязательно запишите пароль себе в защищенное место, при утере пароля изменение параметров репозитория будет невозможным;
- **Среда по умолчанию** - стенд, на котором будет работать репозиторий при запуске отлаживаемых задач. Если требуется запускать задачи на другом стенде, рекомендуется сделать отдельную ветку в Git (например test) и клонировать её в отдельный директорий, подключив новым репозиторием и указав там другую среду по умолчанию;

- **Стенды репозитория** - список стендов, которые используются для разработки, тестирования и промышленной эксплуатации. Рекомендуемый минимальный набор стендов dev (разработка) и prod (промышленная эксплуатация). P.S. После работы с репозиторием в случае командной работы не забывайте закоммитить изменения в Git и сделать push, чтобы другие пользователи смогли их увидеть.

## Импорт или изменение существующего репозитория

Если требуется подключиться к существующему репозиторию, который уже есть в директории (например был получен с Git проекта от другого разработчика), то нажмите в списке репозитория кнопку "Импорт". Для изменения существующего репозитория нажмите кнопку "Изменить":

Create repository

<b>Name</b>	<input type="text" value="MyRepDev"/>
<b>Path</b>	<input type="text" value="/data/repositories/myrep-dev"/>
<b>Password</b>	<input type="text" value="repository password"/>
<b>Env default</b>	<input type="text" value="dev"/>

Envs

+	Value	^
🗑	dev	
	test	
	prod	

Сохранить
Закреть

### Правила импорта или изменения существующих репозитория:

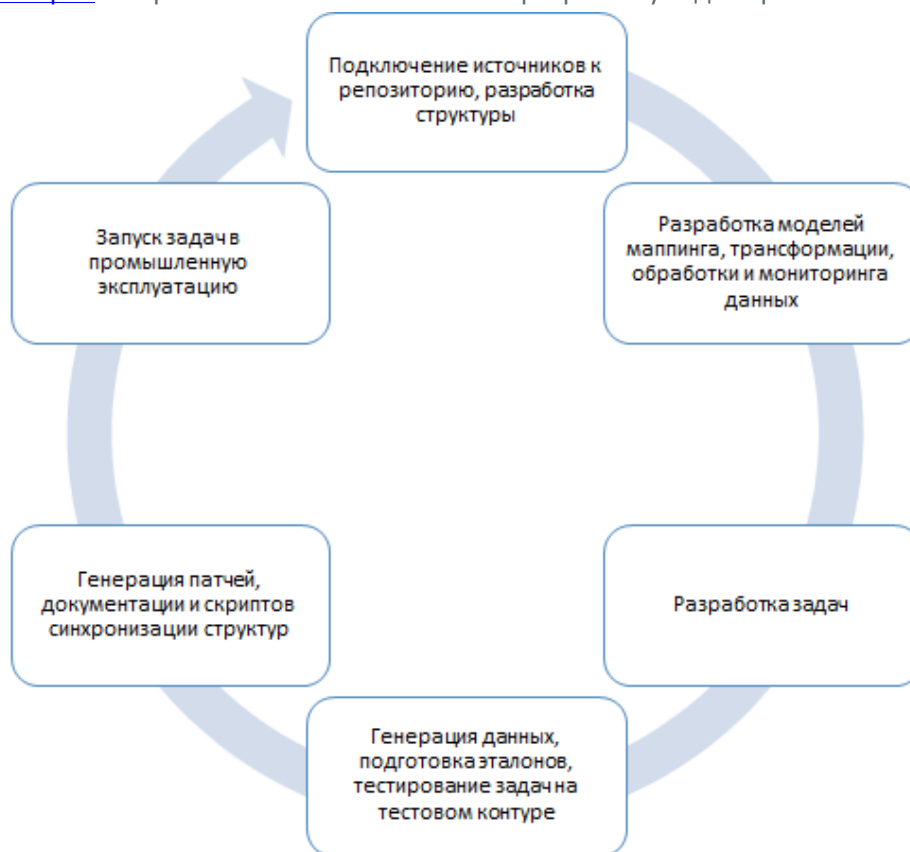
- Если владелец репозитория не передал вам пароль доступа к нему, то при импорте Вы можете задать имя репозитория и выбрать среду по умолчанию;
- Если Вам известен пароль репозитория, то указав этот пароль можно изменить параметры все репозитория. P.S. Если Вы изменили параметры репозитория, не забудьте закоммитить изменения в Git и сделать push, чтобы другие пользователи смогли их увидеть.

## Удаление репозитория

Для удаления репозитория в списке репозитория нужно нажать кнопку "Удалить". При удалении он пропадает с видимости портала, но его файлы остаются на диске. При необходимости можно сделать импорт репозитория, чтобы подключить его обратно.

## Примеры работы

После [создания репозитория](#) в портале в нём можно начинать разработку задач проекта:



## Порядок действий

1. Для работы с источниками требуется описать подключение к ним для каждого стенда, определенного для репозитория;
2. Для обработки данных источников требуется описать их структуру или импортировать её описание с источника, если он поддерживает это. Если описываются новые объекты хранения данных, то после описания потребуется их создать на источнике;
3. Для того, чтобы использовать описанные в репозитории данные источников требуется разработать модели работы с ними, в которых нужно связать между собой объекты источников, их поля и указать правила обработки, которые потребуются для правильной работы скриптов шаблонов;
4. Последовательность шагов действий с помощью вызовов разных скриптов шаблонов и параметров для их выполнения описывается в задачах. Для выбираемых скриптов библиотек шаблонов в документации подробно описаны требования к моделям и правила заполнения параметров. Там же можно для шагов указать дополнительную логику на языке GetI для проверок условий необходимости запуска шагов задачи, инициализации перед выполнением эталонных данных и проверок после выполнения при запуске в тестовой среде;
5. Для облегчения проведения тестирования создаются модели эталонных данных и файлов. После их создания можно вызвать их развертывание в нужных таблицах и директориях перед запуском процессов в задачах и сверить по ним полученный результат после окончания работы;
6. Для переноса разработанных задач репозитория в промышленную эксплуатацию [в панели администрирования](#) для репозитория генерируют zip файл и [загружают его в пакет EasyScheduler](#).

## Правила именования объектов репозитория

Все объекты внутри репозитория именуются по правилам синтаксиса:

`ИмяГруппы1 [ . ИмяГруппы2 [ . ИмяГруппыN ] ] : ИмяОбъектаВнутриГруппы`

В имени объекта обязательно должна присутствовать имя группы, разделенное с ним двоеточием. Если требуется группировку объектов в рамках иерархии групп, в имени группы их следует разделять точкой. В имени объекта не разрешается использовать пробелы и спец символы (решетки, знак доллара, амперсant и т.д.).

## Настройка демонстрационного репозитория DEMO

Переключитесь в [панель администрирования](#), выберите [глобальные переменные](#) и добавьте новую переменную `DEMO_PATH`:

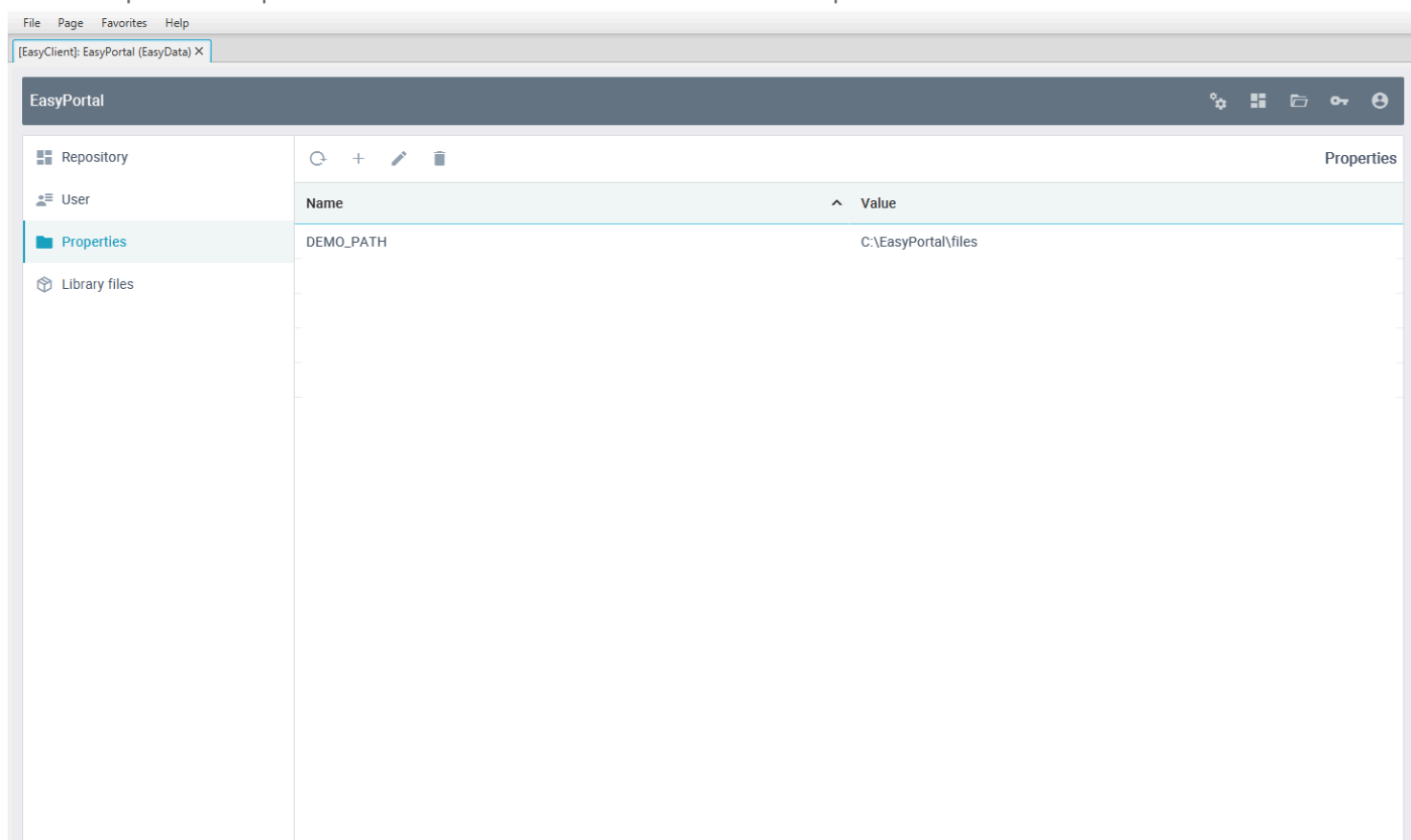
**Create property**

<b>Name</b>	<input type="text" value="DEMO_PATH"/>
<b>Value</b>	<input type="text" value="C:\EasyPortal\files"/>

СохранитьЗаккрыть

- В значении укажите путь к новой директории, где будут храниться файлы примеров;
- Создайте такой директорию в файловой системе.

После сохранения переменной она появится в списке глобальных переменных:



Переключитесь в [репозитории](#). Создайте новый репозиторий *demo*:



## Create repository

Name \*

Path \*

Password \*

envs		libs	properties
+	Key		Default
🗑	dev		<input checked="" type="radio"/>
	prod		<input type="radio"/>

Сохранить

Закреть

- Укажите в пути новый директорий, где будет хранится репозиторий. Портал должен иметь к нему доступ и права на создание и запись директориев и файлов;
- Укажите пароль от 16 до 32 символов (например *"easyportal demo password"*);
- В список стенов репозитория добавьте стенды *dev* и *prod*, выберите *dev* по умолчанию.

Во вкладке "Глобальные переменные" добавьте переменную *"DEMO\_PATH"*:

envs		libs	properties
+	Value		
🗑	DEMO_PATH		
	DEMO_PATH		

После сохранения репозиторий появится в списке:

File Page Favorites Help

[EasyClient]: EasyPortal (EasyData) X

EasyPortal

Repository

User

Properties

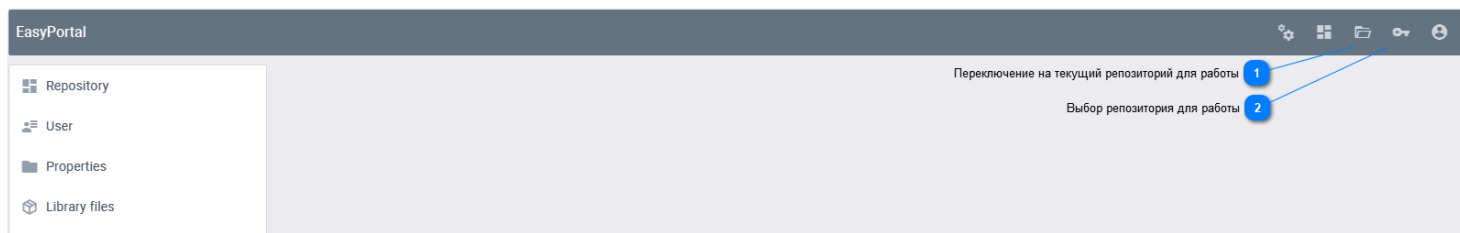
Library files

Name	Path	Env default
demo	C:\EasyPortal\repository	dev

## Копирование данных из таблиц БД в файлы

Для работы с примерами данного раздела выполните шаги из раздела [Настройка демонстрационного репозитория DEMO](#).

Для работы с объектами репозитория в [управляющей панели](#) переключитесь на репозиторий DEMO:



### 1 Переключение на текущий репозиторий для работы



Переключает в режим работы с объектами последнего используемого репозитория.

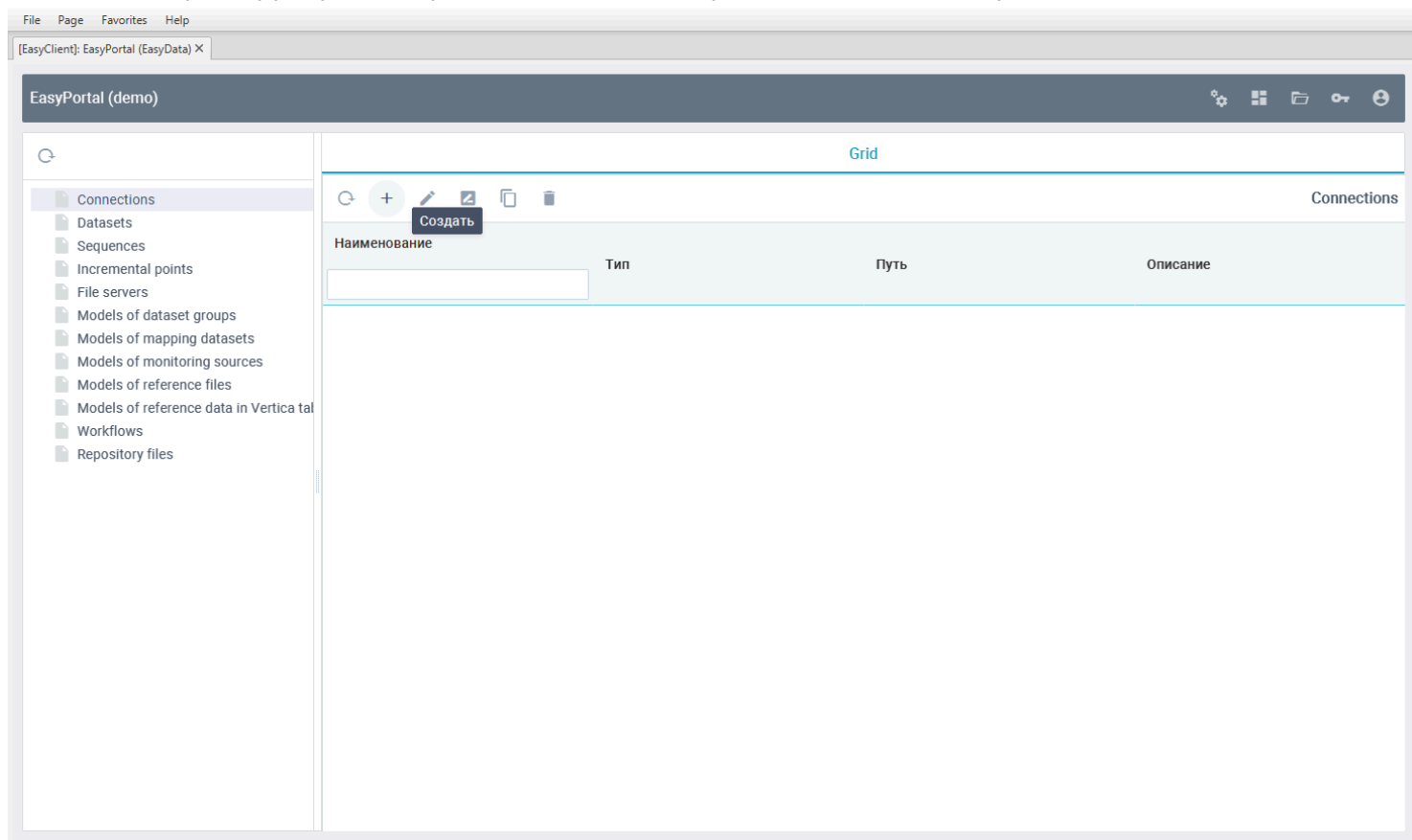
### 2 Выбор репозитория для работы



Выбрать для работы нужный репозиторий и переключиться в режим репозитория для работы с его объектами.

## Создание подключения к файловому источнику CSV

Щелкните в дереве групп репозитория на *Connections* и в гриде нажмите на кнопку *Создать*:



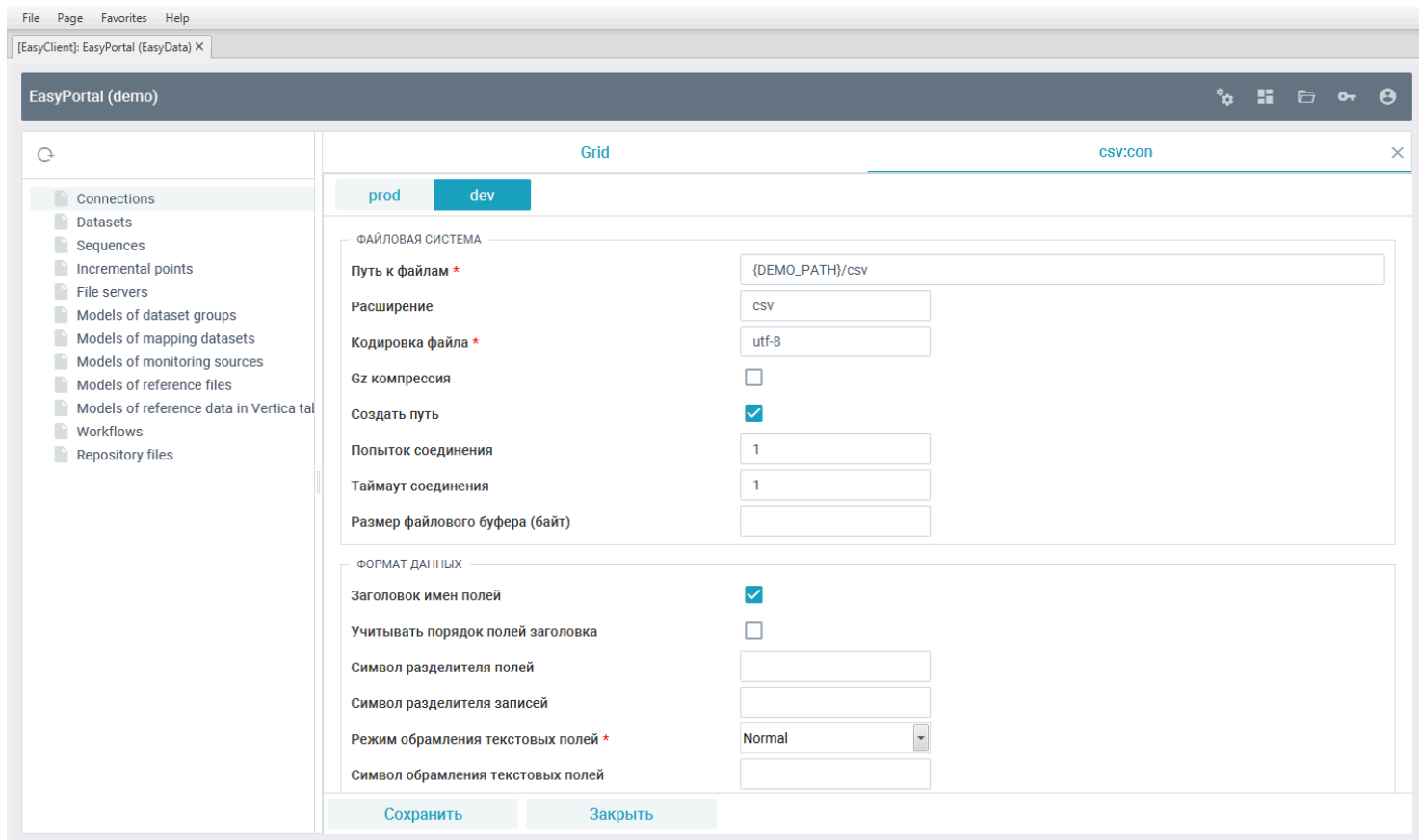
Создайте новое файловое соединение "*csv:con*", где *csv* это название группы в дереве, *con* название соединения в этой группе, выбрав для него тип соединения "*getl.csv.CSVConnection*":

### Создать соединение

**Наименование \***

**Тип \***

После нажатия кнопки "Создать" откроется редактор нового соединения:



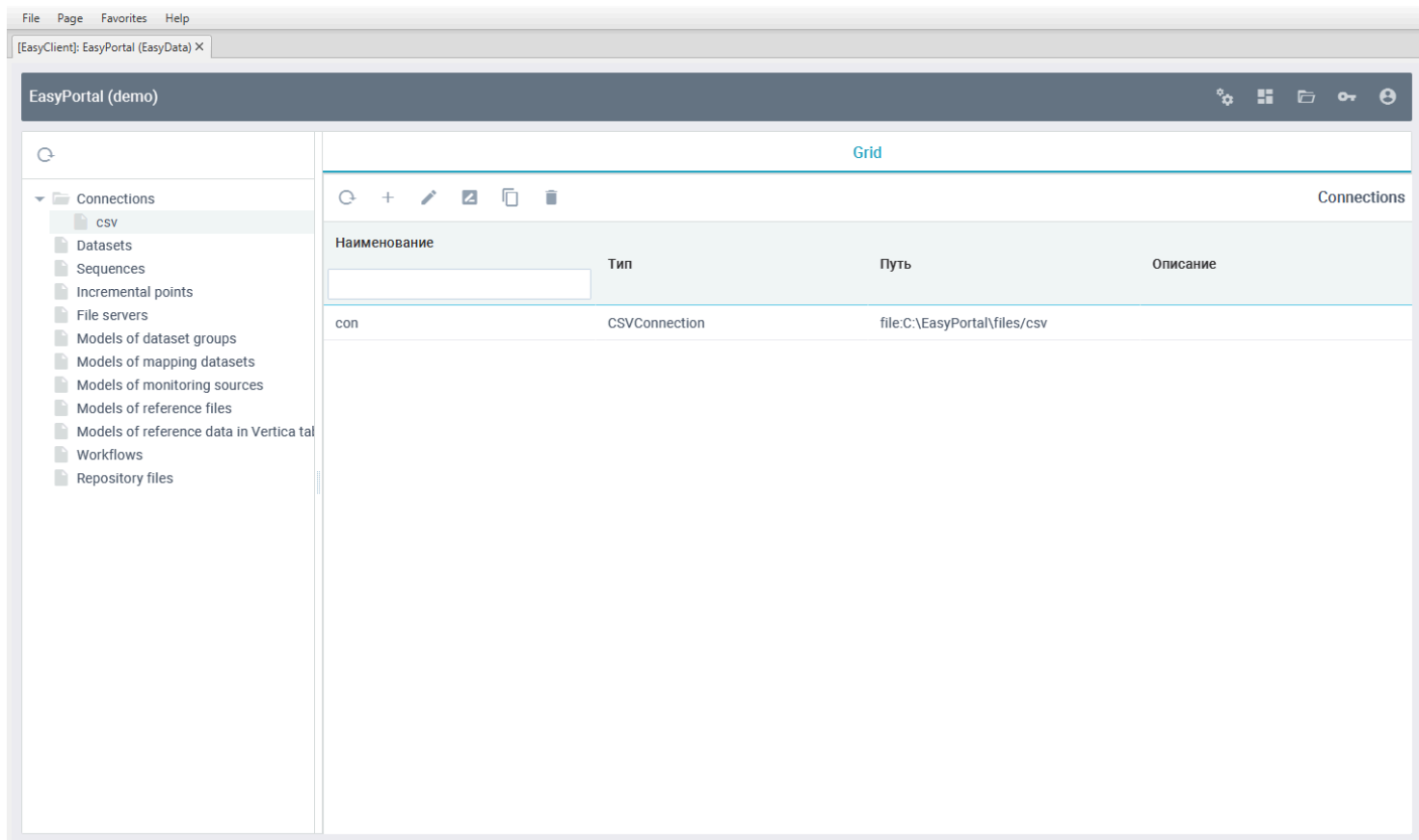
#### Для стенда dev:

- Укажите путь к файлам "{DEMO\_PATH}/csv". Во время работы с файлами источника сюда будет подставляться путь, определенный в глобальной переменной *DEMO\_PATH*;
- Поставьте в расширение "csv". При создании файлов к их именам автоматически это расширение будет добавлено;
- Включите флаг создания пути, чтобы при первом обращении к каталогу он был создан;
- Включите флаг заголовка полей, чтобы первой строкой в CSV файлах шли имена полей;
- Остальные поля оставьте по умолчанию без изменений.

#### Для стенда prod:

Проделайте те же действия, так как файловый источник имеет относительный путь через переменную и при работе на разных серверах зоны разработки и промышленной эксплуатации достаточно проставить разный путь в эту переменную.

Нажмите кнопку "Сохранить" и после успешного сохранения кнопку "Закреть". В дереве репозитория появится новое соединение "csv:con":



P.S. В пути соединения будет показан реальный путь, собранный с учетом выражения из глобальной переменной "DEMO\_PATH". Если поменять значение этой переменной, то путь соединения тоже поменяется.

## Создание подключения к РСУБД H2

По аналогии с CSV соединением создайте новое H2 соединение "h2:con", выбрав для него тип соединения "getl.h2.H2Connection":

### Создать соединение

**Наименование \***

**Тип \***

После нажатия кнопки "Создать" откроется редактор нового соединения:

The screenshot shows the 'EasyPortal (demo)' interface with a sidebar on the left containing a tree view of connections. The main area is titled 'Grid' and shows a configuration window for a connection named 'h2:con' on the 'dev' environment. The configuration is divided into 'ПАРАМЕТРЫ ПОДКЛЮЧЕНИЯ' and 'АУТЕНТИФИКАЦИЯ'. Under 'ПАРАМЕТРЫ ПОДКЛЮЧЕНИЯ', the 'Host' radio button is selected. The 'Хост сервера' field is empty. The 'БД сервера \*' field contains '{DEMO\_PATH}/h2/db'. The 'URL соединения \*' field is empty. The 'Таймаут логина (сек)' field is empty. The 'Количество попыток соединения' field contains '1'. The 'Таймаут соединения (сек)' field contains '1'. The 'Таймаут запросов' field is empty. The 'Автоматический commit' checkbox is unchecked. The 'Размер буфера выборки' field is empty. The 'Уровень изоляции транзакции' dropdown is set to 'Read committed'. The 'Драйвер' field contains 'org.h2.Driver'. The 'Путь для драйвера' field is empty. The 'Включить работу хранимых процедур' checkbox is checked. Under 'АУТЕНТИФИКАЦИЯ', there are empty fields for 'Имя пользователя' and 'Пароль'. At the bottom, there are 'Сохранить' and 'Заккрыть' buttons, and a 'Test' button in the top right corner.

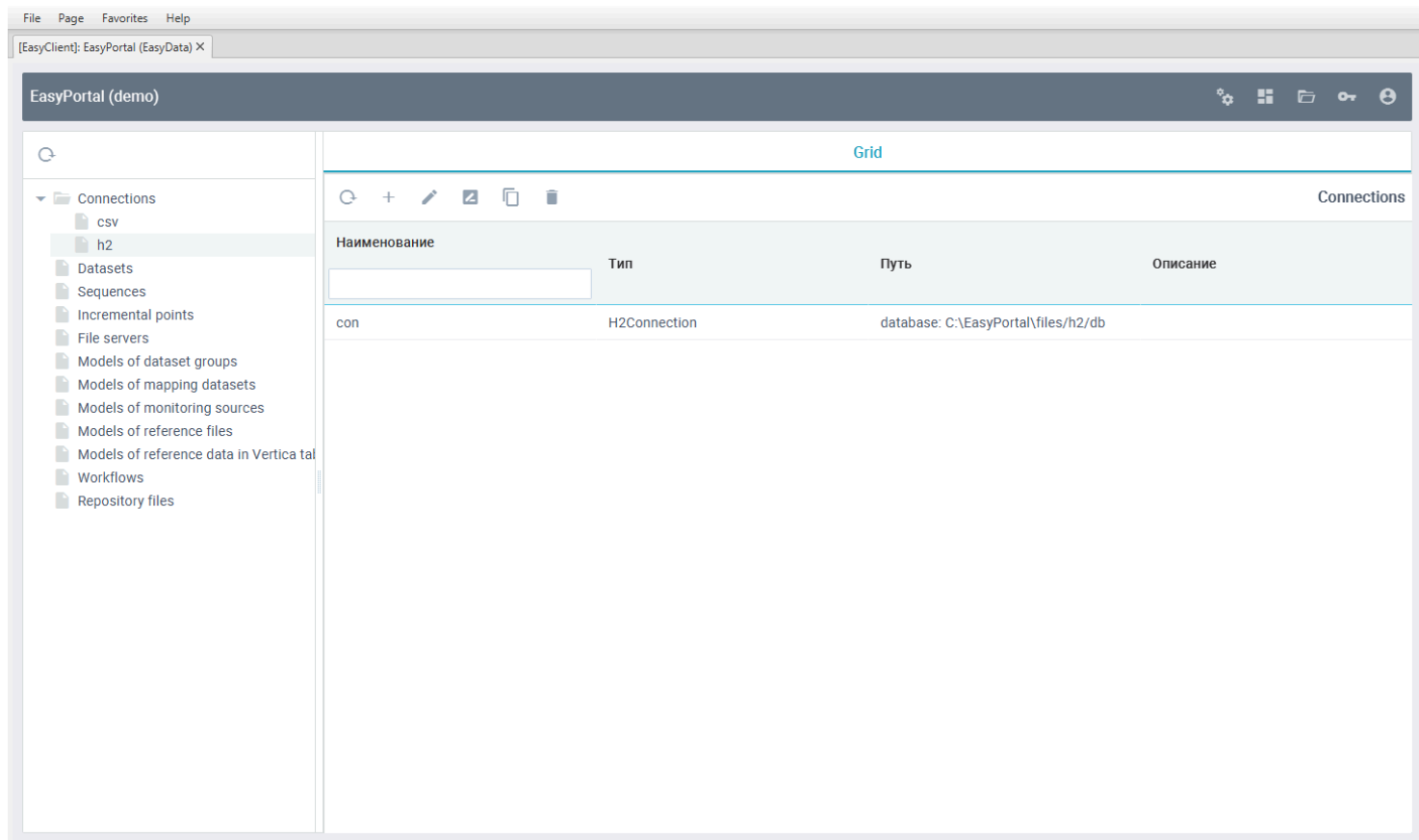
### Для стенда dev:

- Укажите в БД сервера "{DEMO\_PATH}/h2/db". В этом файле будет создана локальная база данных. При работе с БД файлом сюда будет подставлен путь, определенный в глобальной переменной репозитория *DEMO\_PATH*;
- Включите флаг работы хранимых процедур. При выполнении SQL скриптов на этом источнике будет разрешено использовать встроенный в портал язык хранимых процедур Getl;
- Включите флаг заголовка полей, чтобы первой строкой в CSV файлах шли имена полей;
- Укажите в авторизации логин "sa" и пароль "demo";
- Нажмите кнопку "Test", чтобы убедиться, что соединение к БД проходит успешно.

### Для стенда prod:

Проделайте те же действия, что и для стенда dev.

Нажмите кнопку "Сохранить" и после успешного сохранения кнопку "Заккрыть". В дереве репозитория появится новое соединение "h2:con":



P.S. В пути соединения показан реальный путь, собранный с учетом выражения из глобальной переменной "DEMO\_PATH". Если поменять значение этой переменной, то путь соединения в списке соединений тоже поменяется.



## Создание скриптов инициализации БД H2

Зайдите в проводнике в директорий, куда был создан репозиторий и создайте в нем новый директорий "sql":

Имя	Дата изменения	Тип	Размер
getl.lang.sub.RepositoryConnections	20.04.2022 18:40	Папка с файлами	
sql	20.04.2022 18:51	Папка с файлами	
getl-properties.conf	20.04.2022 17:38	Файл "CONF"	1 КБ

В директории "sql" создайте новый файл "init-h2.sql", заполнив его следующим текстом:

```
ЕCHO Инициализация БД H2 ...
```

```
CREATE SCHEMA IF NOT EXISTS DEMO;
```

```
CREATE TABLE IF NOT EXISTS DEMO.DIM (  
    ID INT NOT NULL PRIMARY KEY,  
    NAME VARCHAR(50) NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS DEMO.FACT (  
    DIM_ID INT NOT NULL,  
    DT TIMESTAMP NOT NULL,  
    VALUE NUMERIC(12, 2)  
);
```

```
ЕCHO Готово.
```

После сохранения файл с SQL скриптом станет доступным в репозитории. Для того, чтобы увидеть новые директории и файлы репозитория, обновите дерево групп репозитория кнопкой обновить сверху и выберите в дереве Repositories Files/sql:

The screenshot shows the EasyPortal (demo) interface. On the left, a tree view shows the repository structure under 'Repository files', with 'sql' selected. A blue circle with the number '2' points to the 'Файлы репозитория' (Repository files) section. In the main area, a 'Grid' view shows a table with the following data:

Name	Type	Size
init-h2.sql	20.04.2022 18:59	288 b


A blue circle with the number '1' points to the 'Обновление дерева репозитория' (Refresh repository tree) button in the top left of the main area.

## 1 Обновление дерева репозитория



Перечитывает структуру групп объектов репозитория

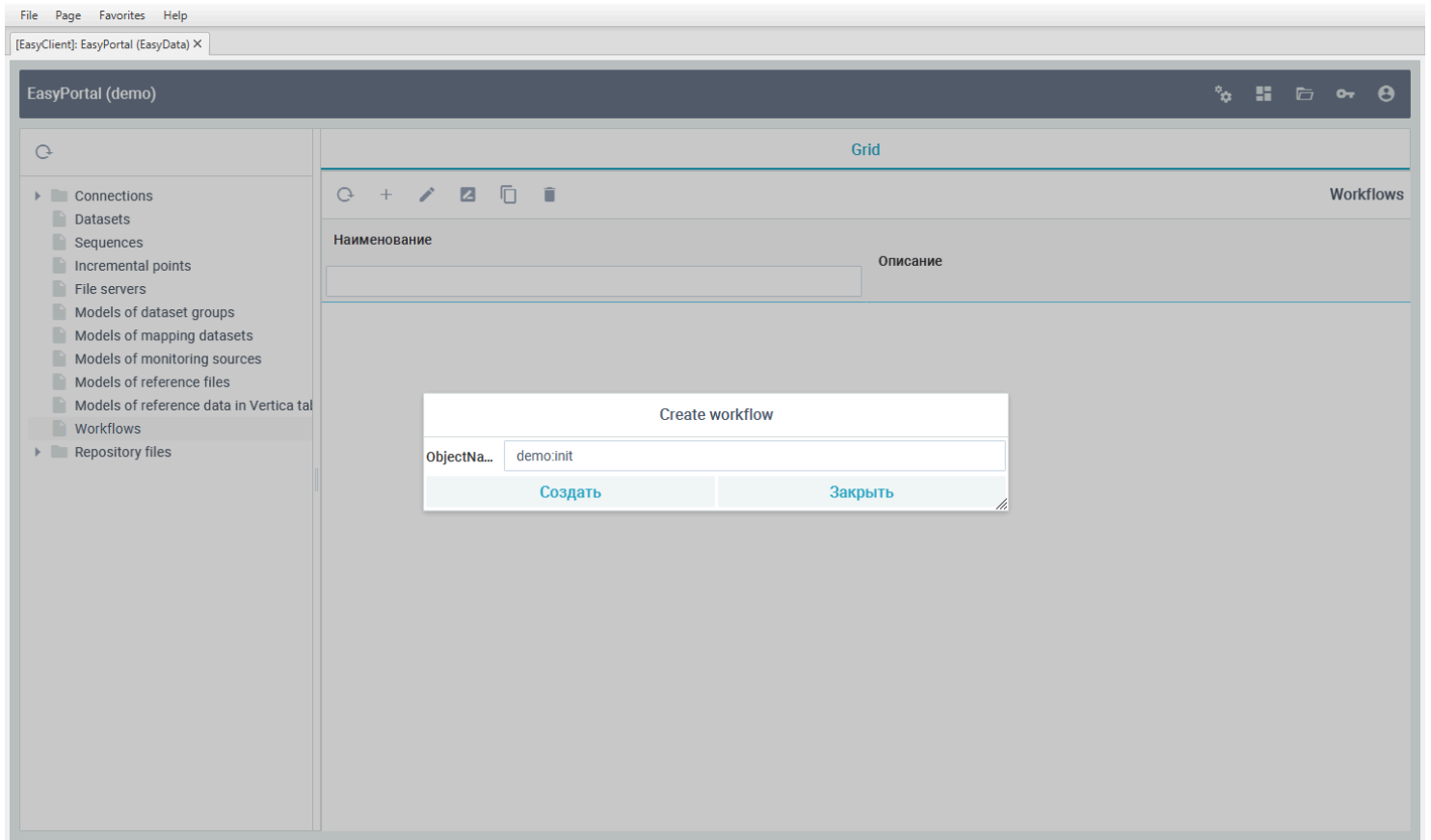
## 2 Файлы репозитория

 Repository files

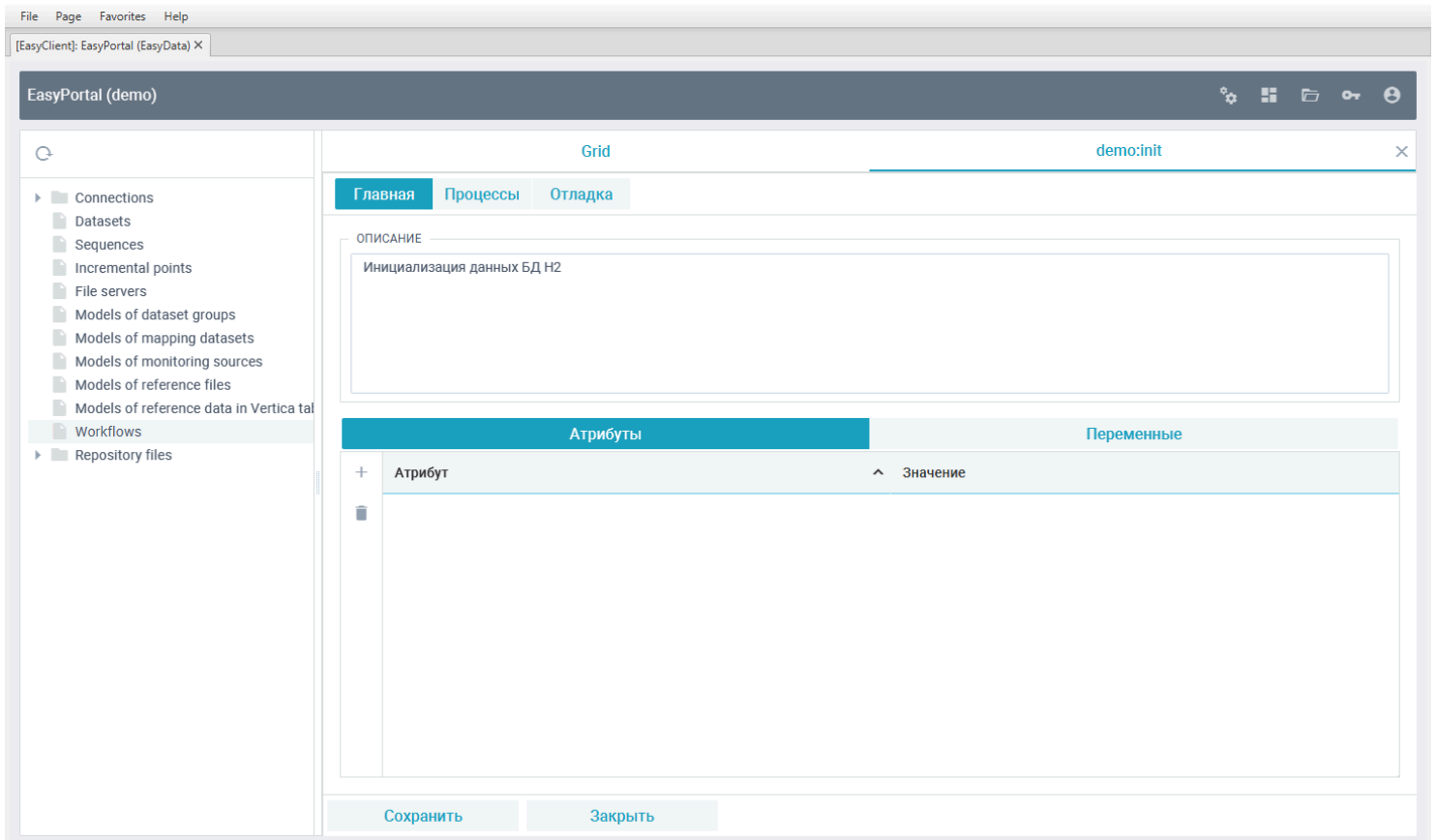
Дополнительные внешние директории и файлы репозитория

## Создание задачи инициализации БД H2

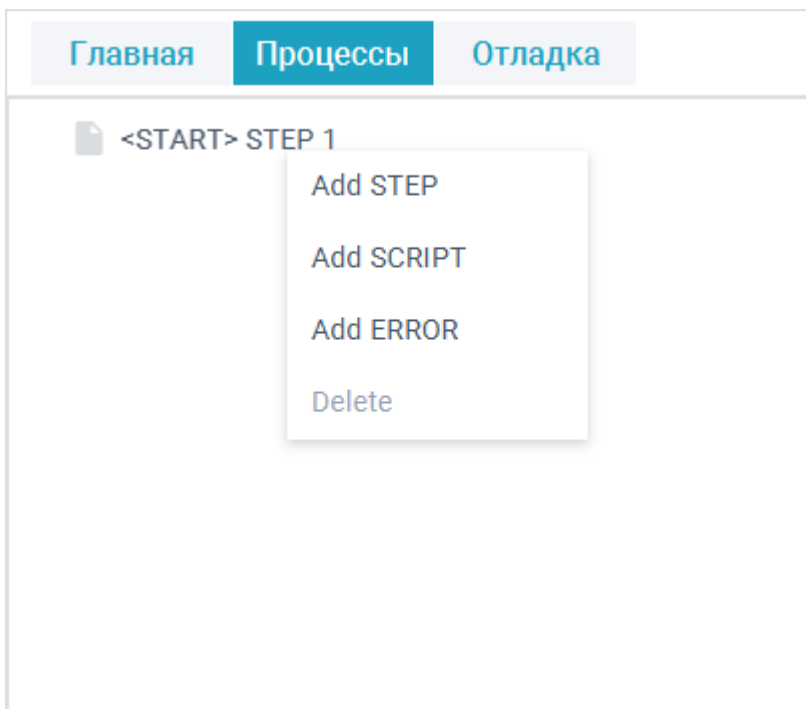
Встаньте в дереве объектов репозитория на раздел "Workflows" и в гриде нажмите "Создать":



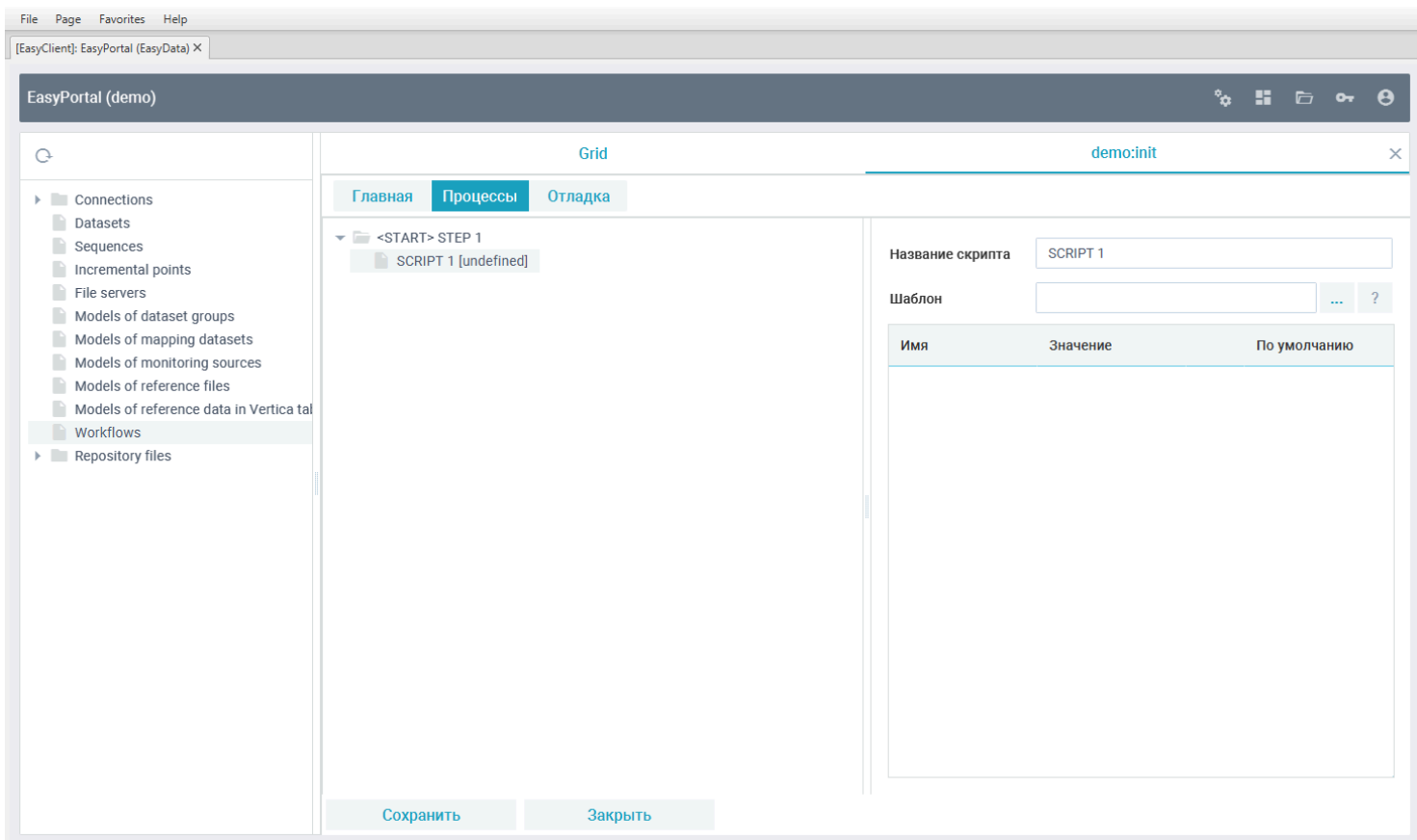
Введите имя "demo:init". Задача будет создана в группе demo с именем init. После нажатия кнопки "Создать" откроется редактор задачи:



На главной вкладке задачи можно вбить описание задачи, которое потом отобразится в списке задач. Переключитесь на вкладку "Процессы" и на главном шаге задачи "STEP 1" нажмите правой кнопкой мышки:



Выберите "ADD SCRIPT". В главный шаг будет добавлен скрипт для выполнения:



Нажмите кнопку с тремя точками справа от поля "Шаблон", чтобы выбрать нужный шаблон процесса задачи и в открывшемся окне выберите шаблон RunSql:

LibrarieClasses

- ▼ EASYLOADER
- ▼ ru
- ▼ easydata
- ▶ files
- ▶ generator
- ▼ rdbms
- ▶ CopyToCsv
- ▶ CopyToTables
- ▶ LoadToBufferTables
- ▶ RunSql
- ▶ vertica
- ▶ TFM

## Шаблон RunSql

Запускает указанный файл с SQL скриптом на указанном JDBC соединении. Рекомендуется использовать для выполнения действий над загруженными в БД данными, расчета витрин и других действий в рамках одной БД.

### Параметры шаблона

- **connection**: JDBC соединение в репозитории, обязательный параметр;
- **login**: используемый логин соединения. Если не указан, то используется текущий;
- **script\_file\_path**: путь к файлу SQL скрипта, обязательный параметр;
- **code\_page**: кодовая страница файла скрипта, по умолчанию utf-8;
- **allow\_procedure\_statements**: использовать парсер операторов языка хранимых процедур Getl. Если параметру не задано значение, то оно берется из extensionForSqlScripts свойства соединения;
- **debug\_mode**: включить печать выполняемых команд в консоль, по умолчанию выключено.
- **sql\_vars**: значения для переменных скрипта.

### Переменные скрипта sql\_vars

Если внутри скрипта используются переменные в фигурных скобках, значения для них можно задать несколькими способами:

- Для workflow:
  - В параметре скрипта sql\_vars в формате "имя\_переменной1: 'значение1', имя\_переменной2: 'значение2', ...";
  - В переменной модели sql\_vars в аналогичном формате;
- При запуске job или workflow:
  - Параметром командой строки в формате vars.sql\_vars="имя\_переменной1:

Выбрать
Закреть

После нажатия кнопки "Выбрать" в скрипте задачи появятся параметры выбранного процесса:

- В параметр "connection" вбейте имя ранее созданного соединения "h2:con" или выберите его с помощью кнопки параметра с тремя точками;
- В параметр "sql\_file\_path" вбейте путь к созданному файлу SQL скрипта репозитория "repository:/sql/init-h2.sql". Так как файл находится в репозитории, то для него не нужно писать абсолютный путь, достаточно указать, что он находится в "repository:/".

Главная Процессы Отладка

<START> STEP 1  
SCRIPT 1 [ru.easydata.rdbms.RunSql]

Название скрипта SCRIPT 1

Шаблон ru.easydata.rdbms.RunSql

Имя	Значение	По умолчанию
allow_procedure_stat		
code_page		utf-8
commit_on_complete		true
connection	h2:con	
debug_mode		false
login		
script_file_path	repository:/sql/init-h2	
sql_vars	repository:/sql/init-h2.sql	

Сохранить Закреть

Теперь можно выполнить созданную задачу на *dev* стенде. Для этого переключитесь во вкладку "Отладка" и нажмите кнопку запуска задачи:

File Page Favorites Help

[EasyClient]: EasyPortal (EasyData) X

EasyPortal (demo)

Grid demo:init

Главная Процессы Отладка

Логирование Тестирование  Переменные запуска

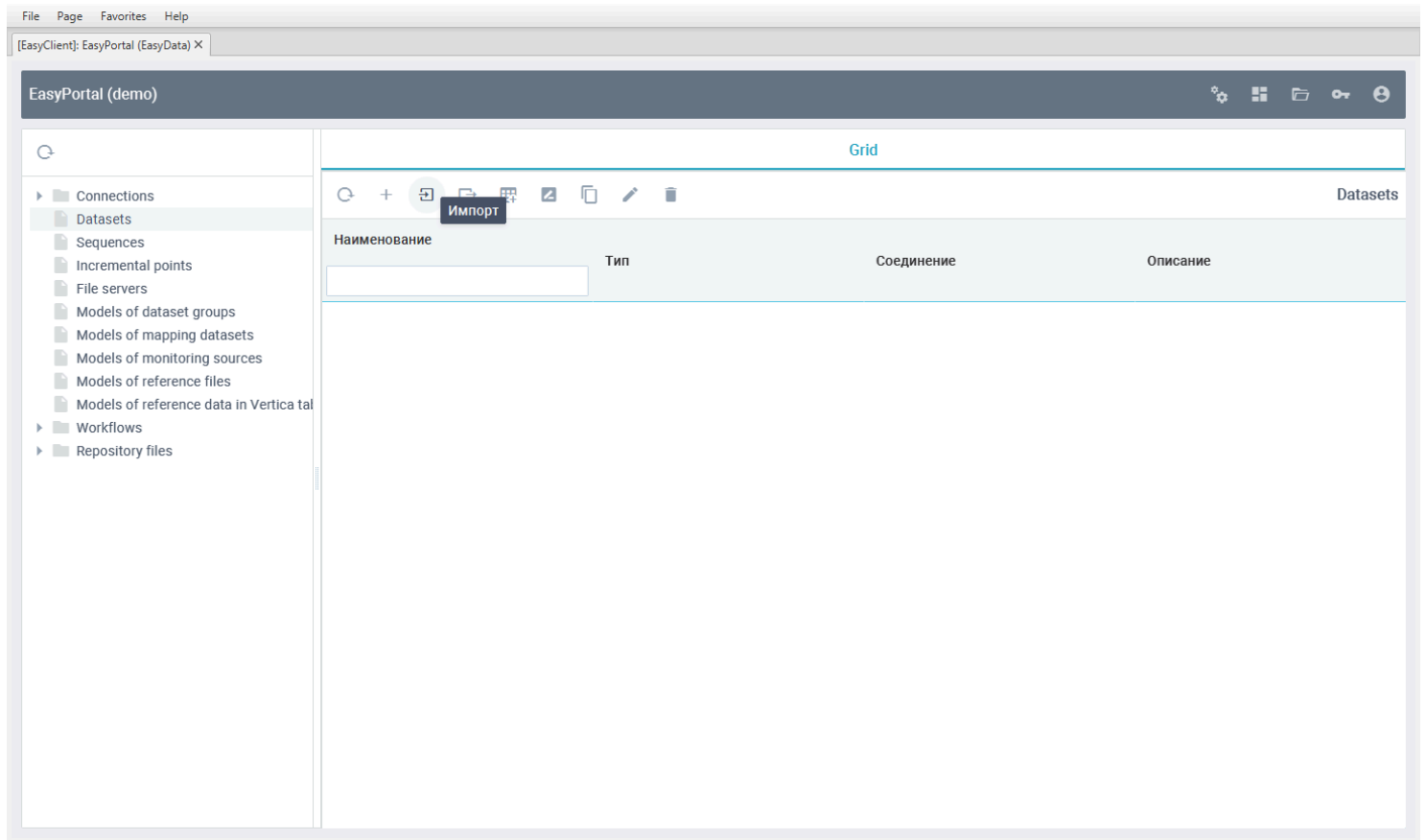
12	2022-04-20 20:43:11 [ FINEST]: Start "STEP 1" step ...
13	2022-04-20 20:43:11 [ FINEST]: Execute script "SCRIPT 1" by class ru.easydata.rdbms.RunSql with step "STEP 1" ...
14	2022-04-20 20:43:12 [ FINE]: *** EasyLoader: выполнение SQL скрипта
15	2022-04-20 20:43:12 [ FINE]: соединение: database: C:\EasyPortal\files\h2\db
16	2022-04-20 20:43:12 [ FINE]: логин: sa
17	2022-04-20 20:43:12 [ FINE]: файл скрипта: repository:/sql/init-h2.sql
18	2022-04-20 20:43:12 [ FINE]: commit по завершении: Да
19	2022-04-20 20:43:12 [ FINE]: язык хранимых процедур: включен
20	2022-04-20 20:43:12 [ FINE]: вывод команд в консоль: выключен
21	2022-04-20 20:43:12 [ FINE]: Инициализация БД H2 ...
22	2022-04-20 20:43:12 [ FINE]: Готово.
23	2022-04-20 20:43:12 [ INFO]: --- Execution 1 steps from workflow "##demo_init##" model completed successfully
24	2022-04-20 20:43:12 [ INFO]: ### Workflow demo:init completed successfully.
25	2022-04-20 20:43:12 [ INFO]: ### Job stop
26	Workflow finish

Сохранить Закреть

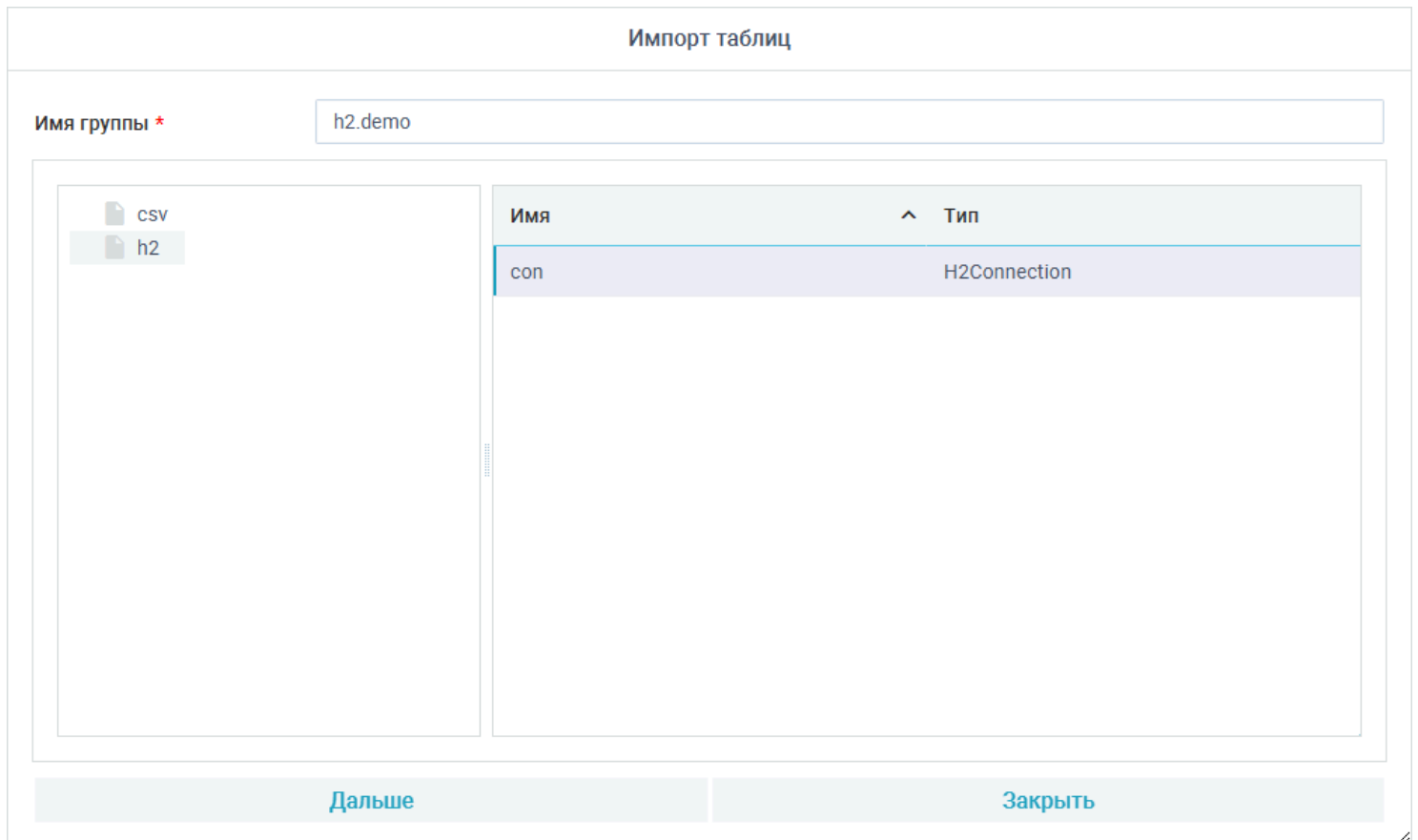
После завершения работы джоба в БД будут созданы схема и две таблицы.

## Импорт созданных таблиц из БД H2 в репозиторий

Для того, чтобы работать в портале с наборами данных (датасетами), требуется сформировать их описание в метаданных каталога объектов. Для существующих объектов РСУБД возможно автоматическое получение их структуры с источника. Для этого в дереве объектов встаньте на "Datasets" и нажмите кнопку "Импорт":



Введите имя группы "h2.demo", куда будут импортироваться таблицы и выберите соединение "h2:con":



Нажав кнопку "Далее", выберите все таблицы со схемы "DEMO":

**Импорт таблиц**

---

**Таблицы**

- ▼  DEMO
  - DIM
  - FACT

НазадИмпортЗакреть

После нажатия кнопки "Импорт" таблицы появятся в репозитории в группе h2, подгруппе demo:

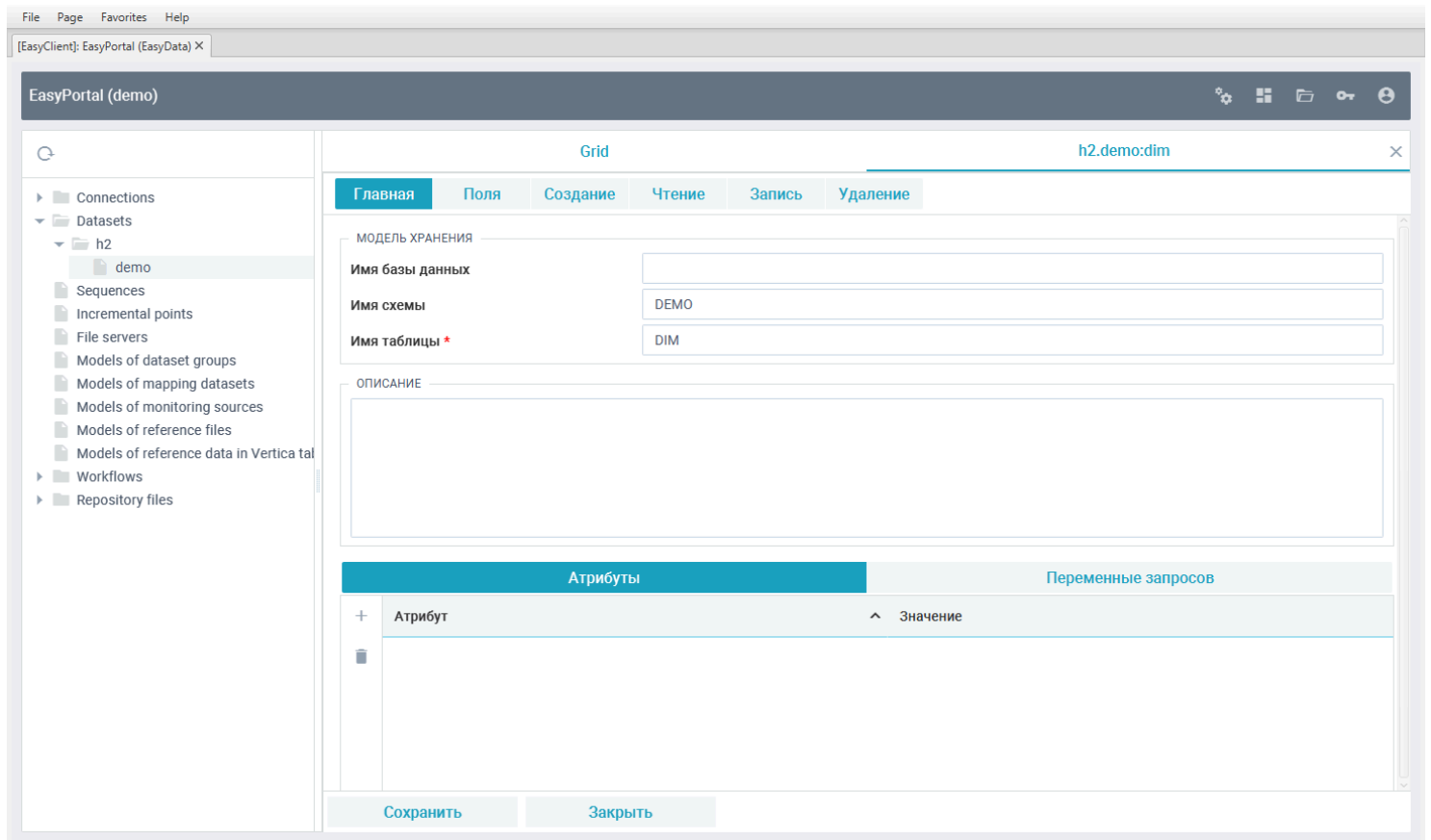
The screenshot shows the EasyPortal (demo) interface. On the left is a navigation tree with 'demo' selected under 'h2'. The main area displays a table of datasets.

Наименование	Тип	Соединение	Описание
dim	H2Table	h2:con	
fact	H2Table	h2:con	

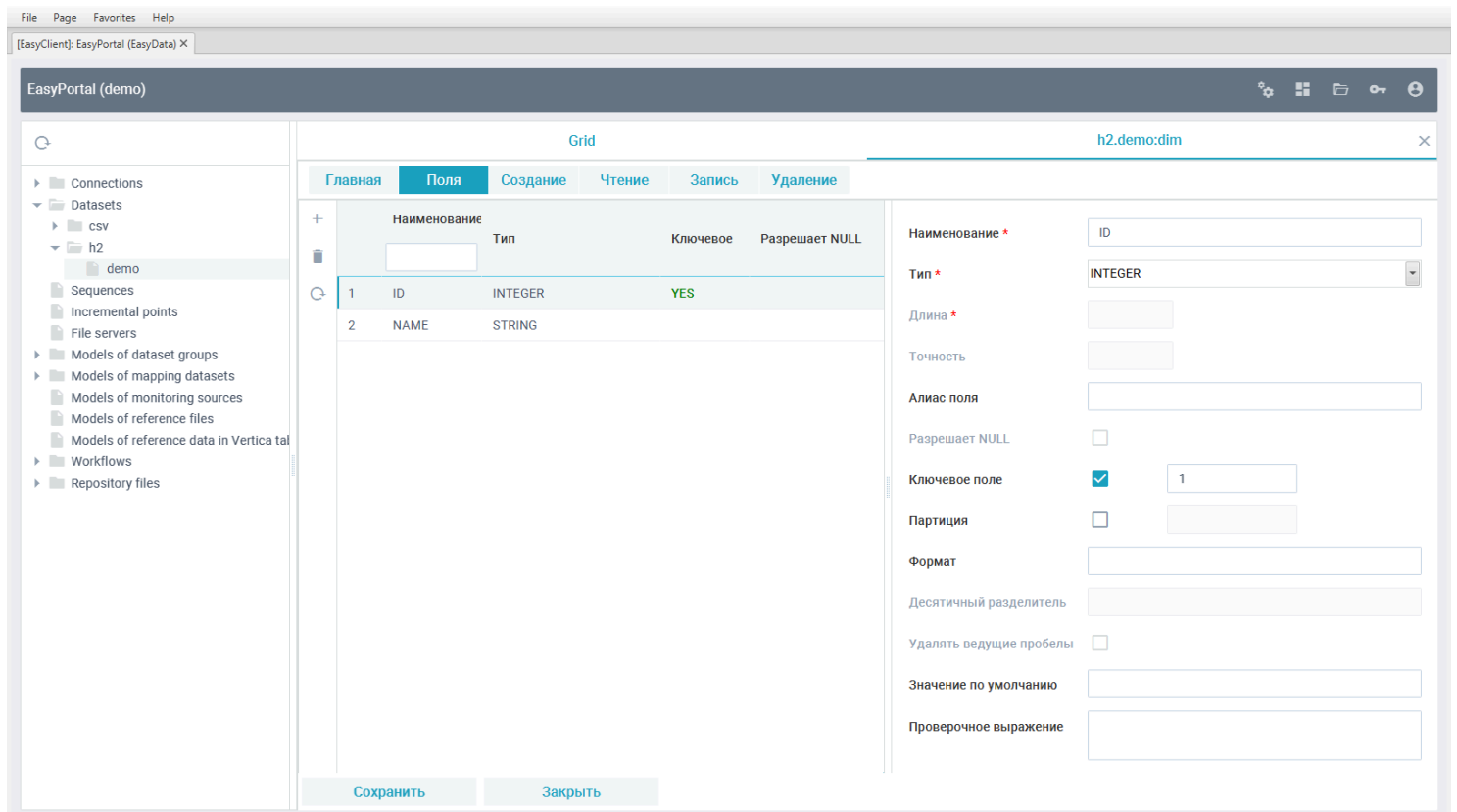


P.S. При импорте таблиц в качестве имен объектов репозитория берутся их имена, определенные в БД источника. Если в именах содержатся запрещенные в правилах именования объектов репозитория символы, они заменяются на подчеркивание.

Для проверки корректности импорта два раза щелкните на таблице "dim" или нажмите кнопку "Изменить" на панели грида. Откроется редактор таблицы:



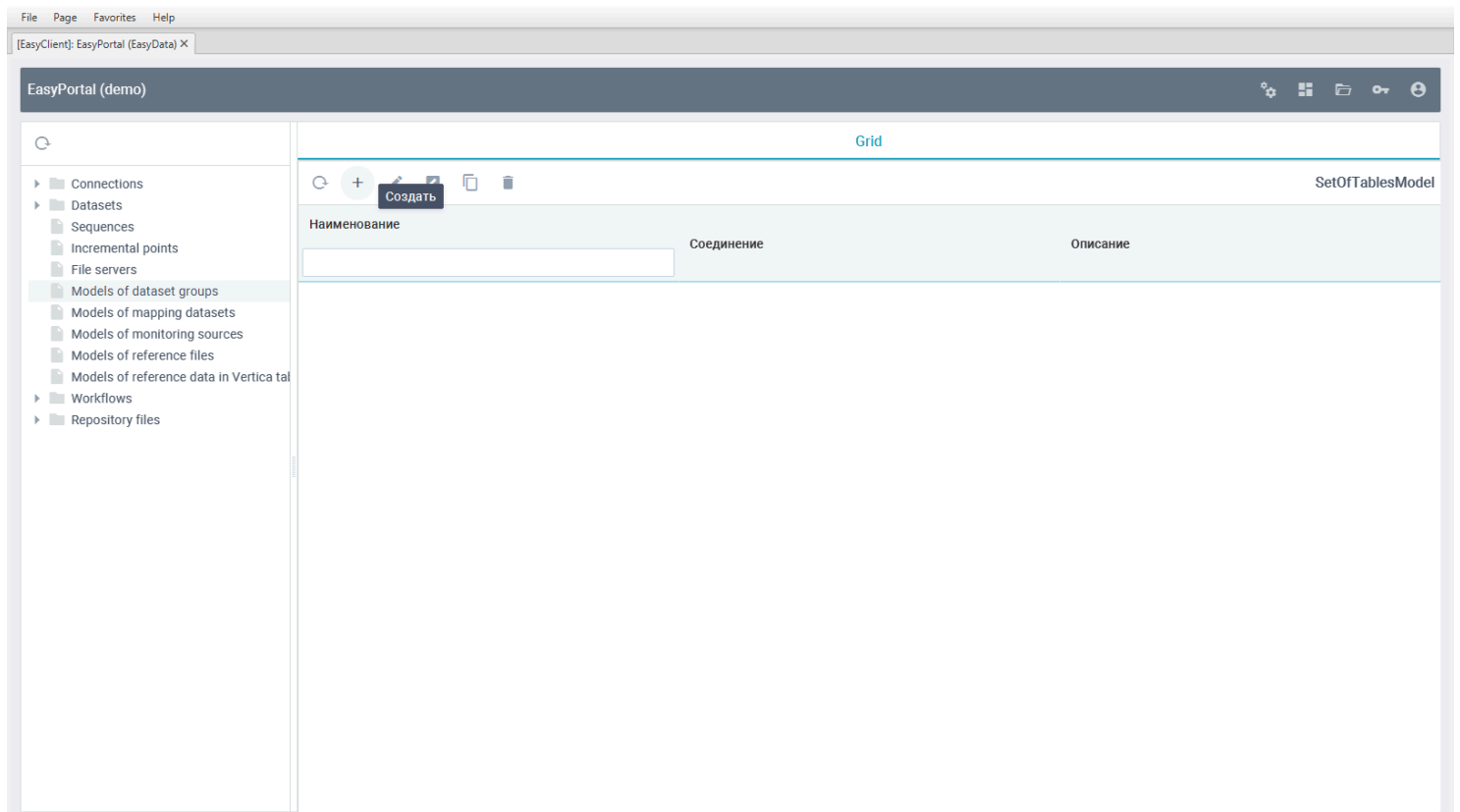
Во вкладке полей таблицы будут перечислены все полученные с БД поля:



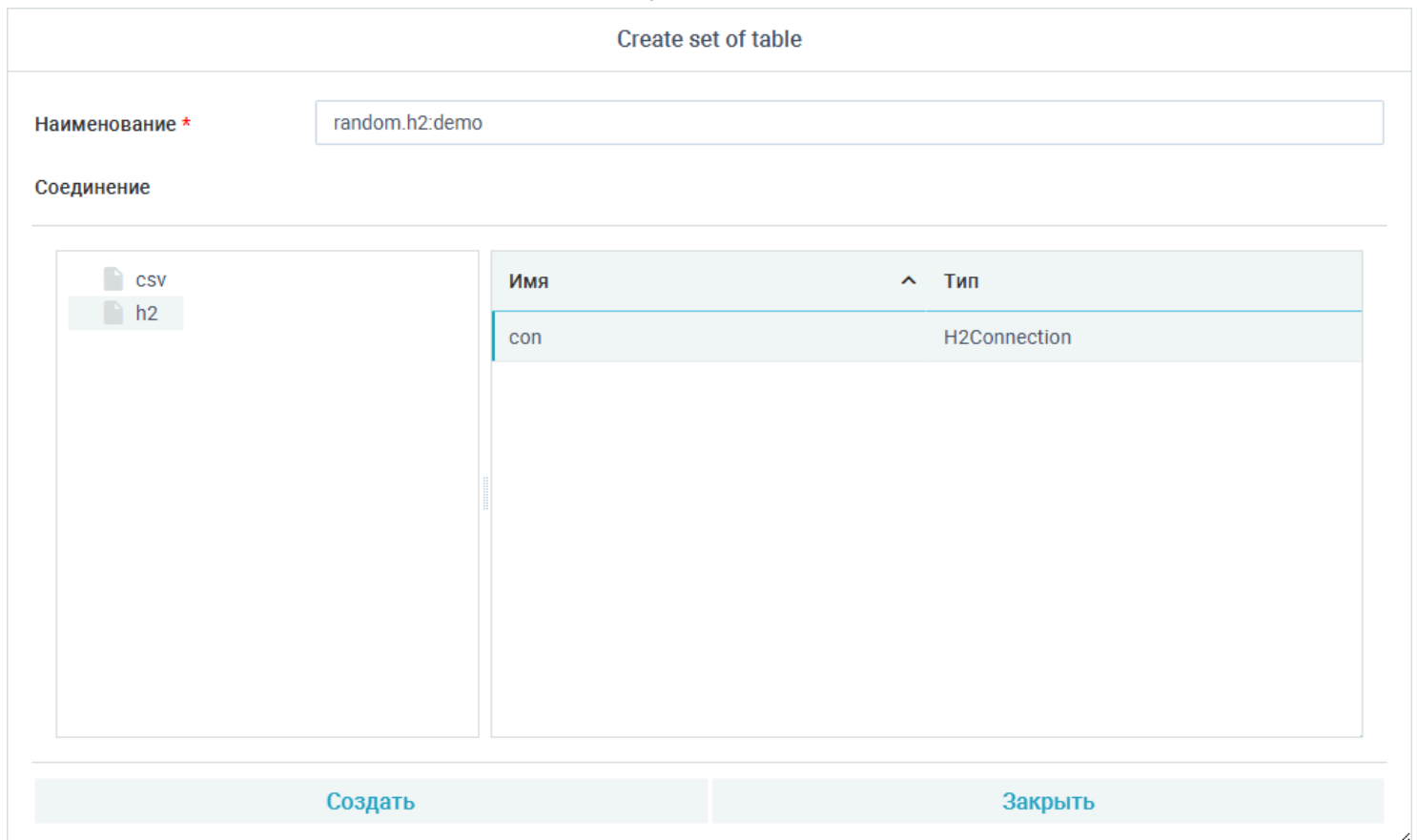


## Создание модели генерации данных в H2 таблицы

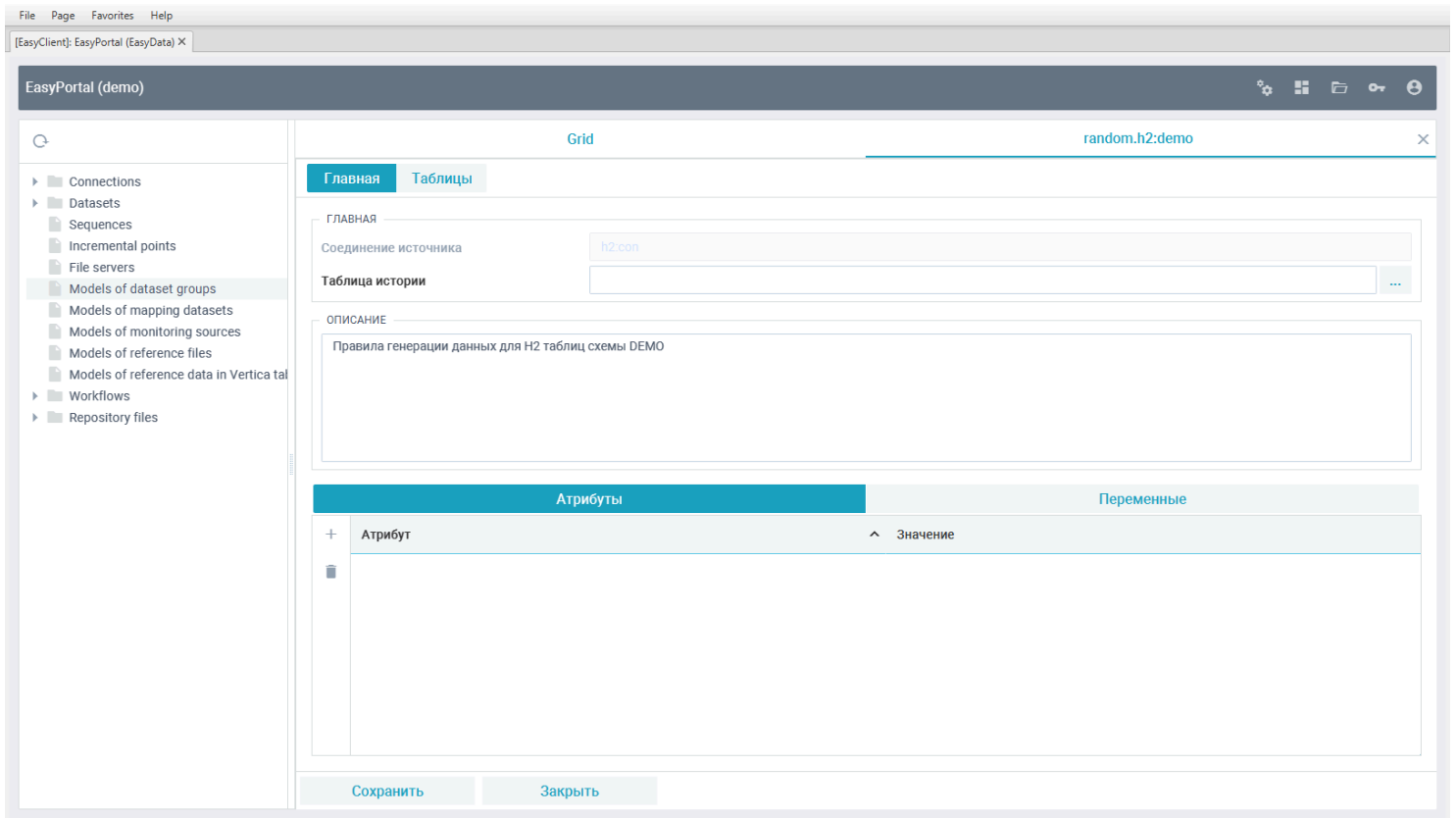
Для генерации случайных данных в подключенные к репозиторию таблицы требуется создать модель трансформации с перечислением таблиц и правил их заполнения. Для этого в дереве объектов переключитесь на группу "Model of dataset groups" и нажмите кнопку "Создать":



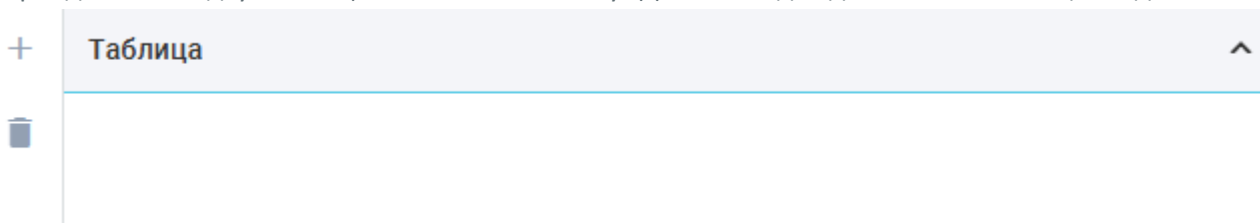
Введите имя новой модели "random.h2:demo" и выберите соединение для модели "h2:con":



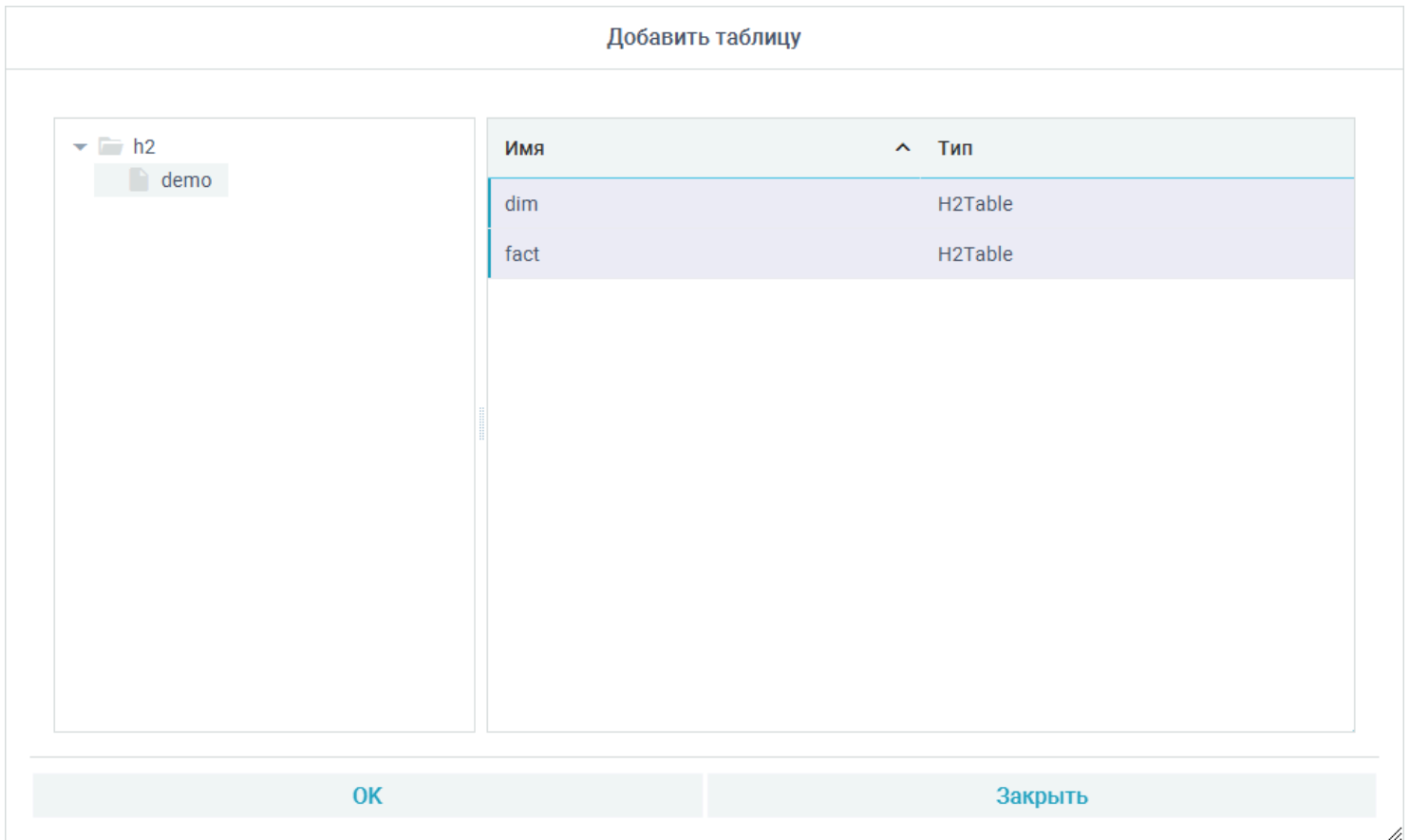
После нажатия кнопки "Создать" откроется окно редактирования модели:



Перейдите в вкладку "Таблицы" и нажмите кнопку "Добавить" для добавления таблиц в модель:

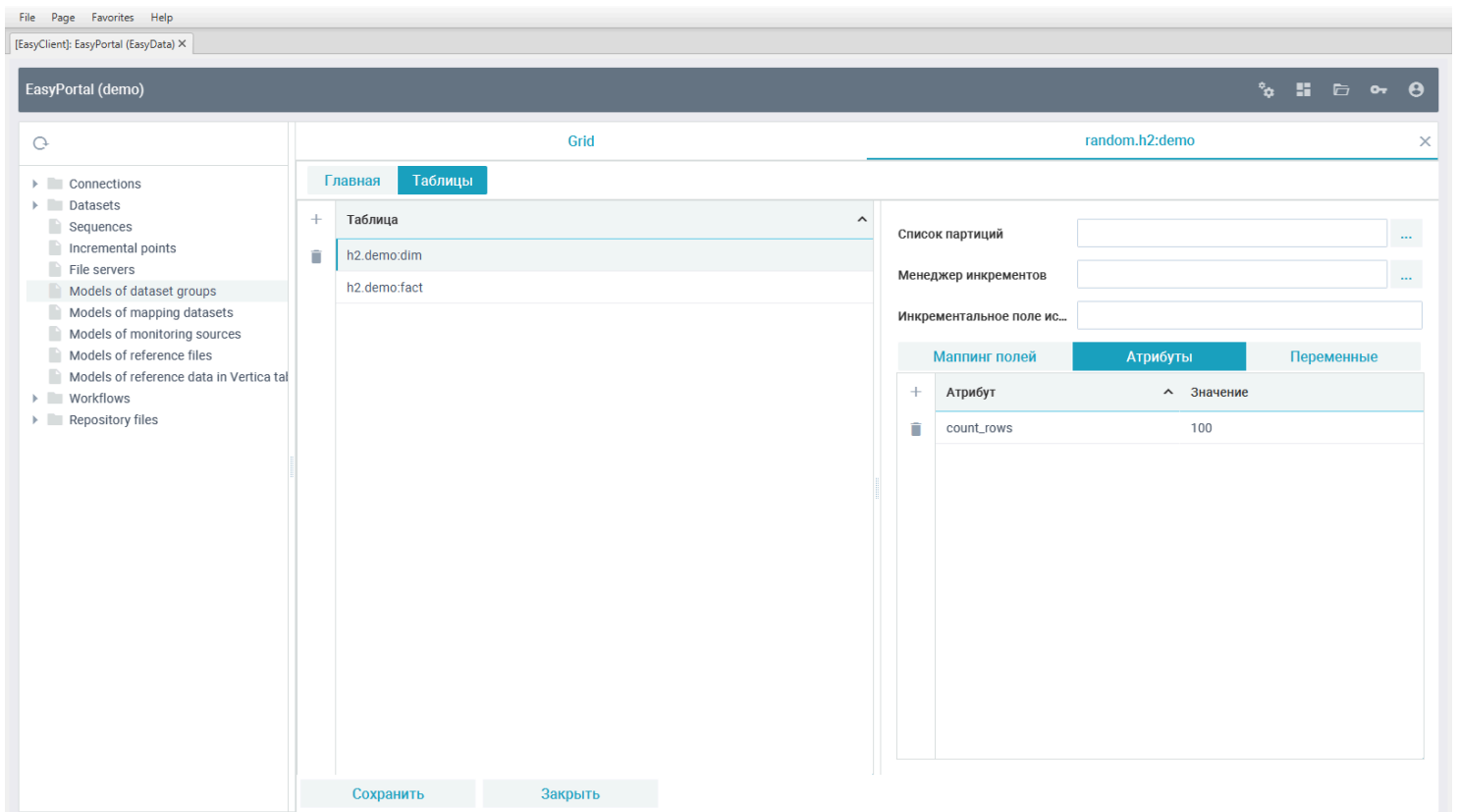


Выберите все таблицы из группы "h2.demo":



P.S. Для выбора множества объектов по очереди можно зажать Ctrl и выбирать левым щелчком мыши. Для выбора всех объектов можно встать на первый в списке и зажав на Shift выбрать последний объект в списке левым щелчком мыши.

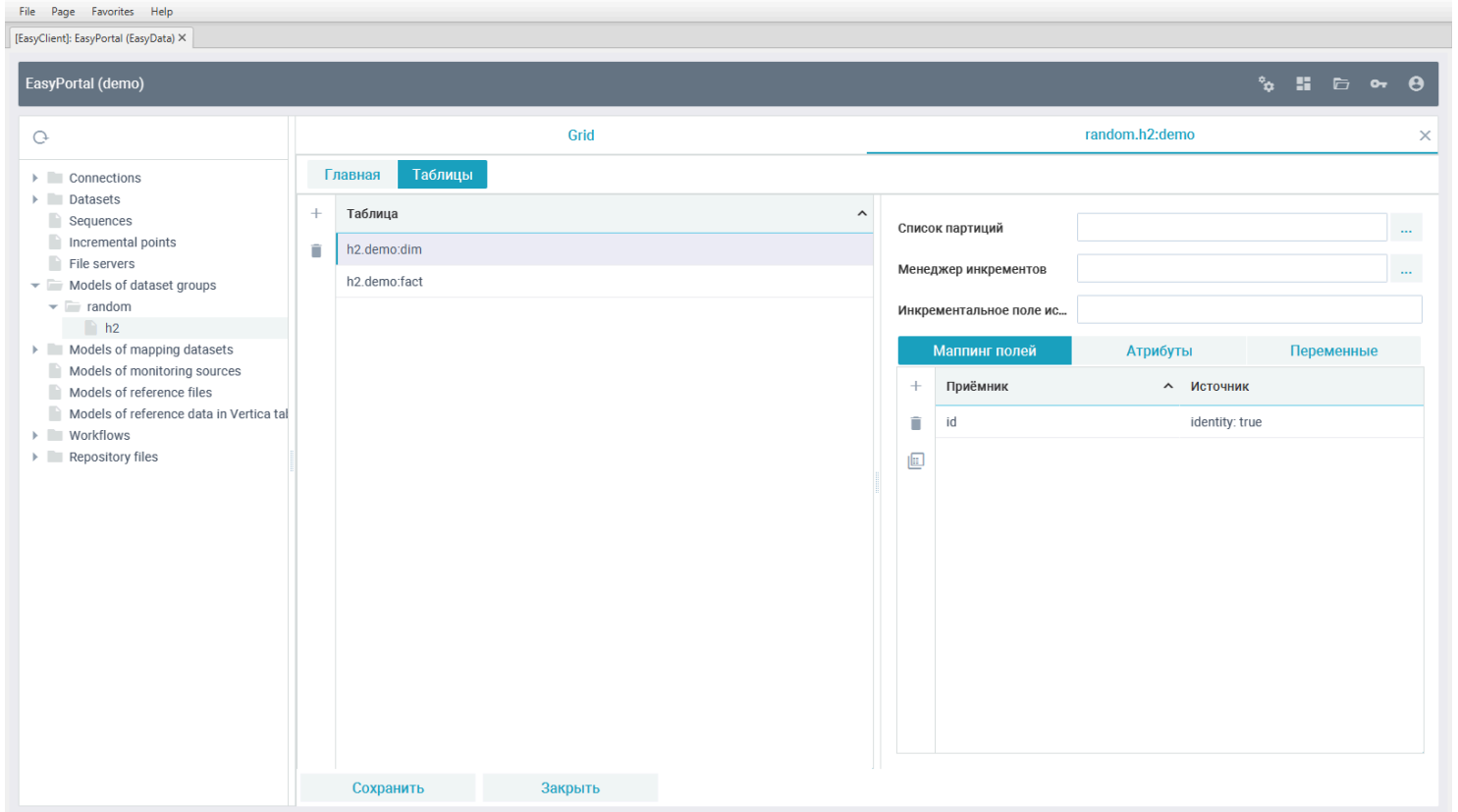
После нажатия кнопки "OK" таблицы будут добавлены в модель. Встаньте в списке таблиц на таблицу "h2.demo:dim", переключитесь справа на вкладку "Атрибуты" и добавьте следующие атрибуты:



Расшифровка атрибутов:

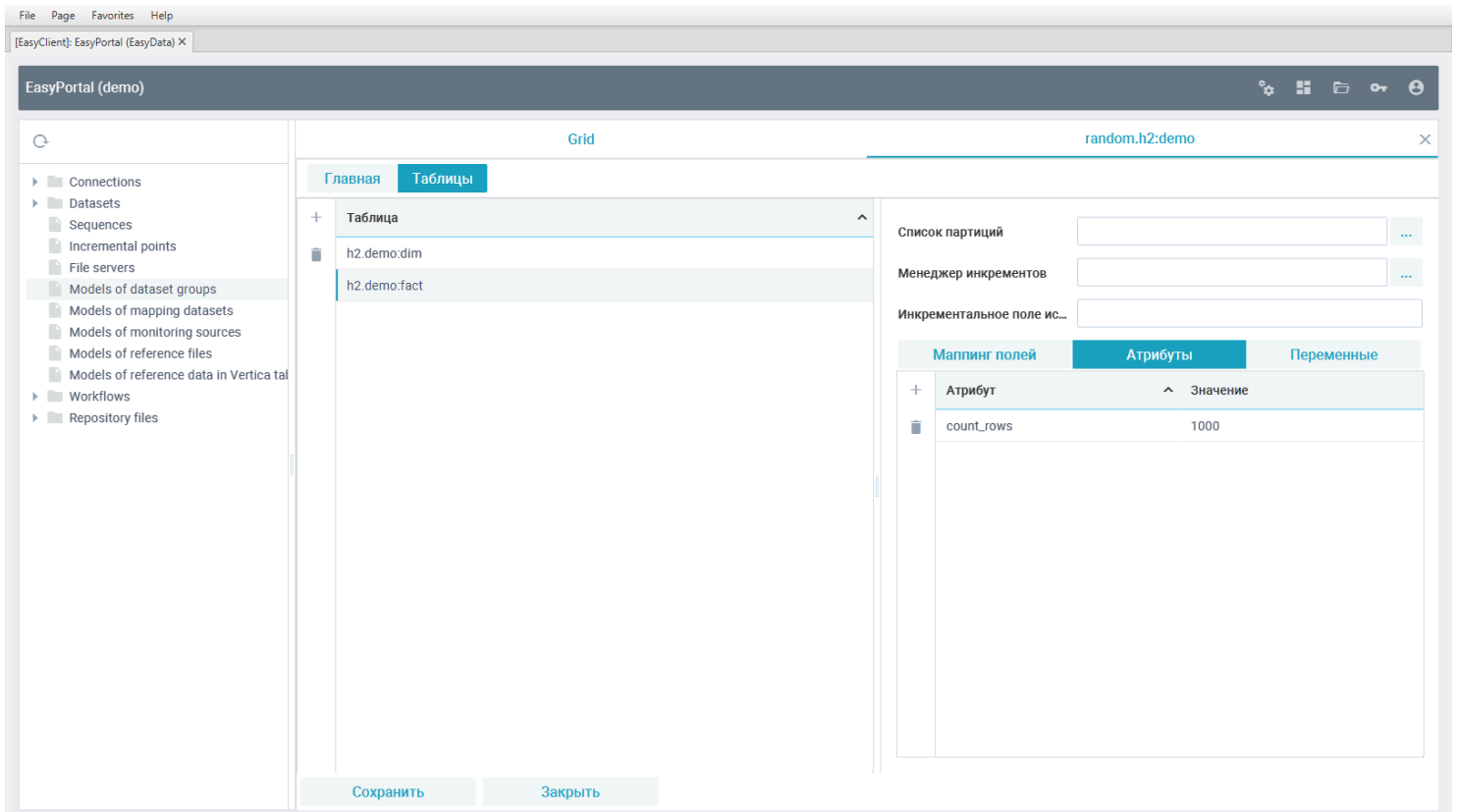
- count\_rows - количество генерируемых записей в таблице

Переключитесь на маппинг полей и добавьте для поля "ID" правило "identity: true":

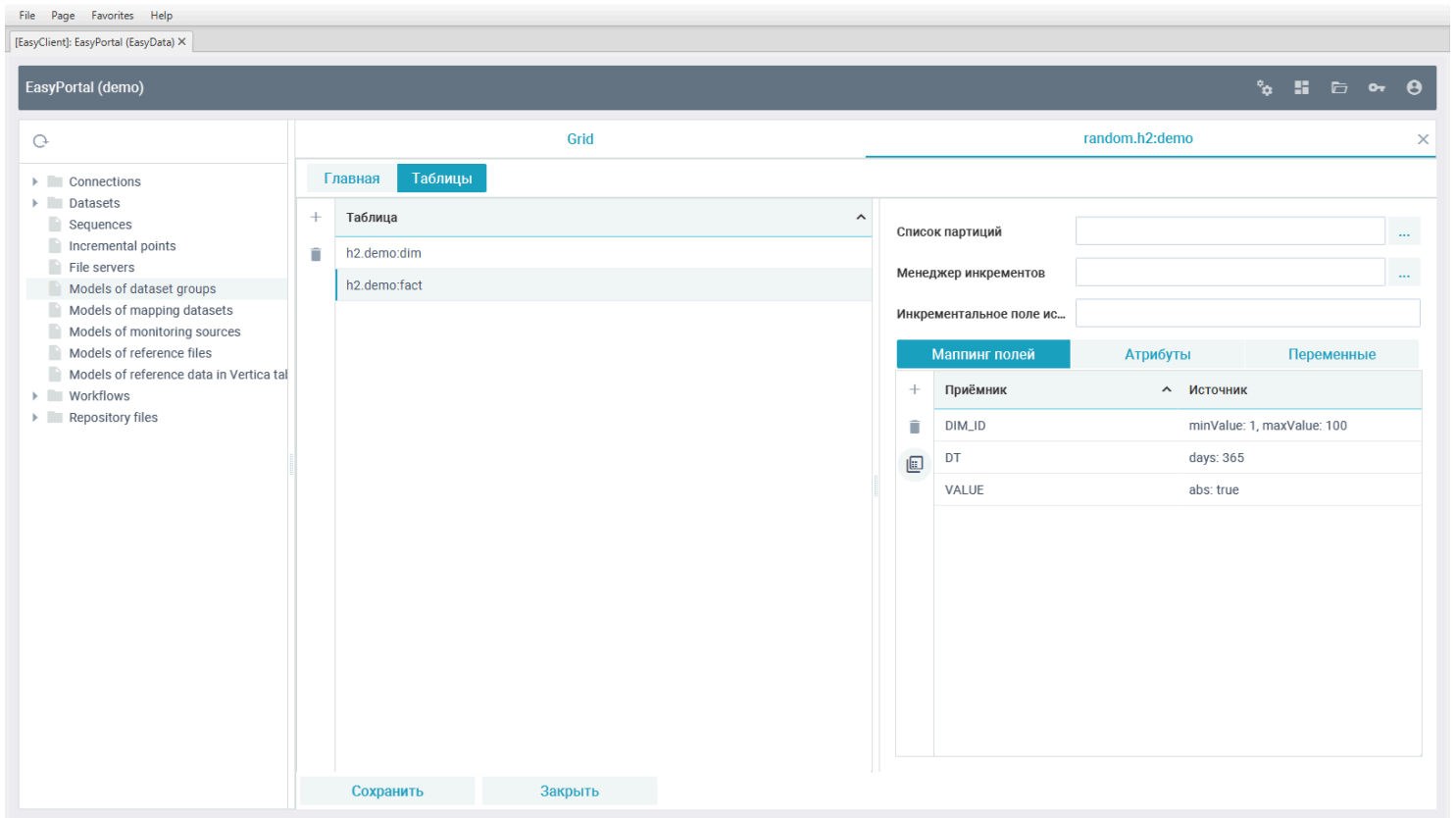


P.S. Включение identity для поля означает, что оно будет заполняться не случайными числами, а последовательными, начиная с единицы.

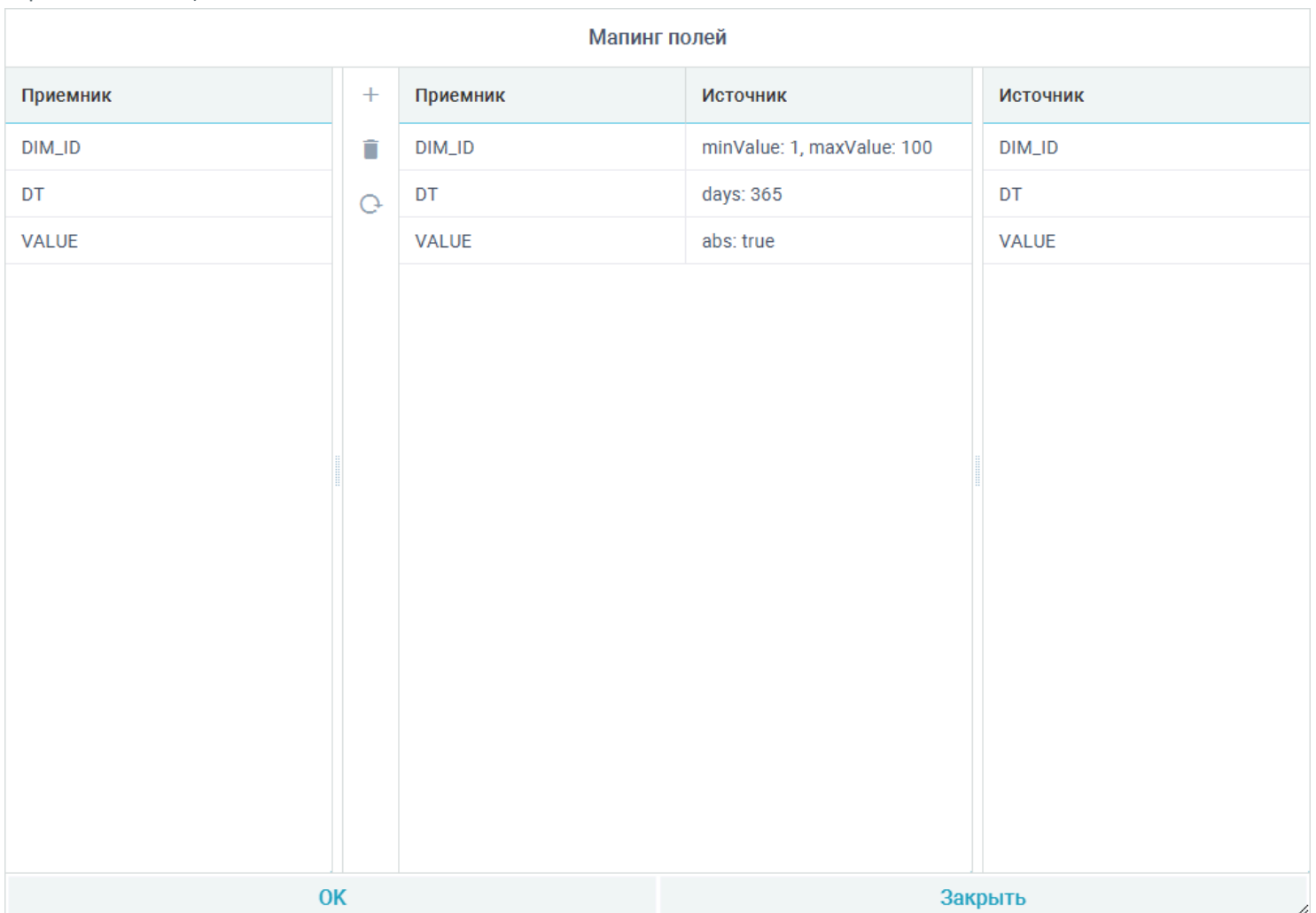
Встаньте на таблицу "h2.demo:fact" и во вкладке "Атрибуты" добавьте следующие атрибуты:



Переключитесь на вкладку "Маппинг" и введите следующие правила для полей таблицы:



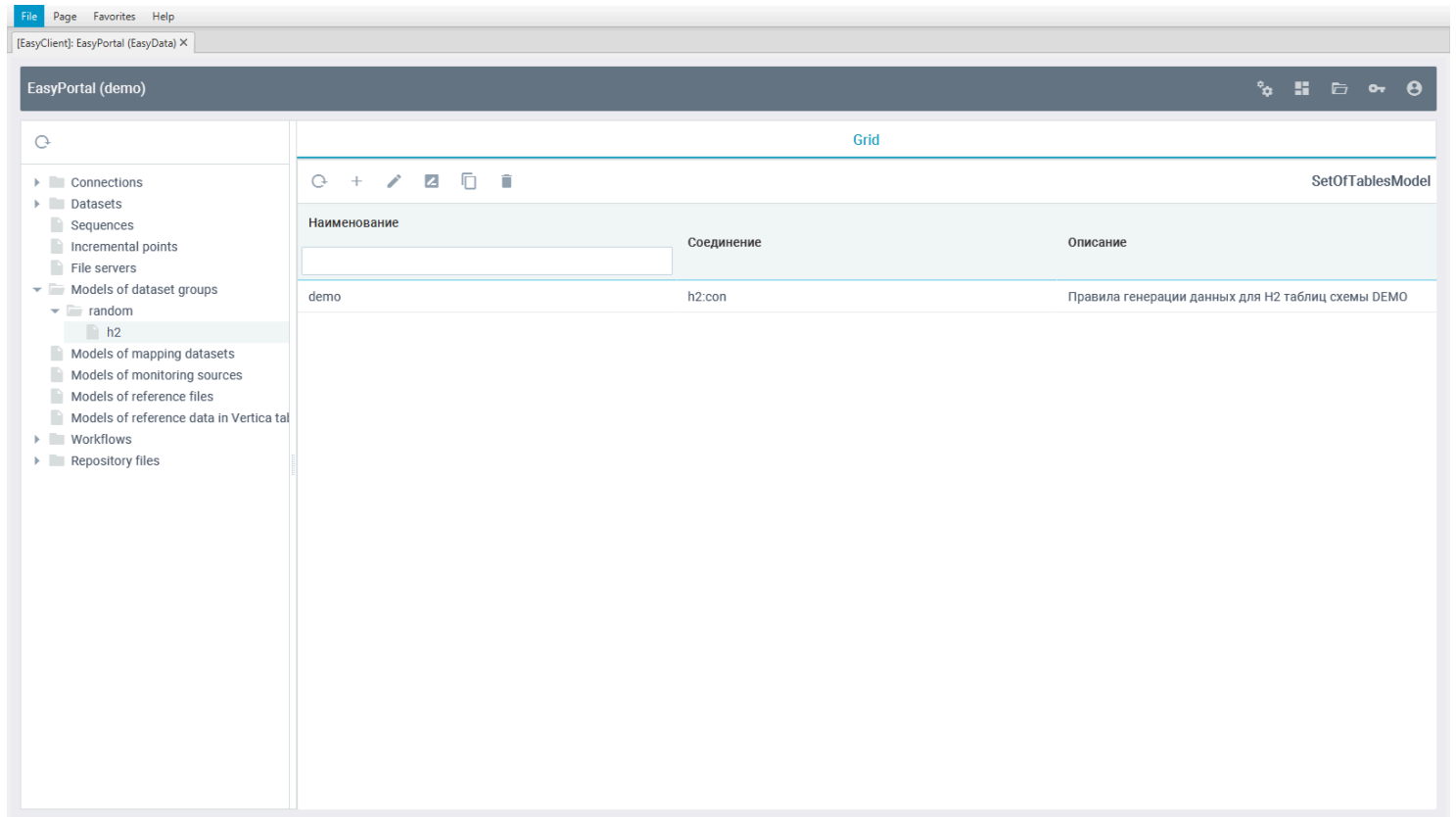
Для более удобного заполнения маппинга полей можно нажать на кнопку редактора маппинга (выделена на скриншоте выше):



Расшифровка правил генерации в маппинге полей:

- `minValue`: генерировать случайное число не менее указанного (указываем 1, чтобы вписаться в предел значений ID справочника);
- `maxValue`: генерировать случайное число не более указанного (указываем 100, чтобы вписаться в предел значений ID справочника);
- `days`: генерировать дату и время в пределах количества указанных дней от текущей даты;
- `abs`: генерировать только положительные числа.

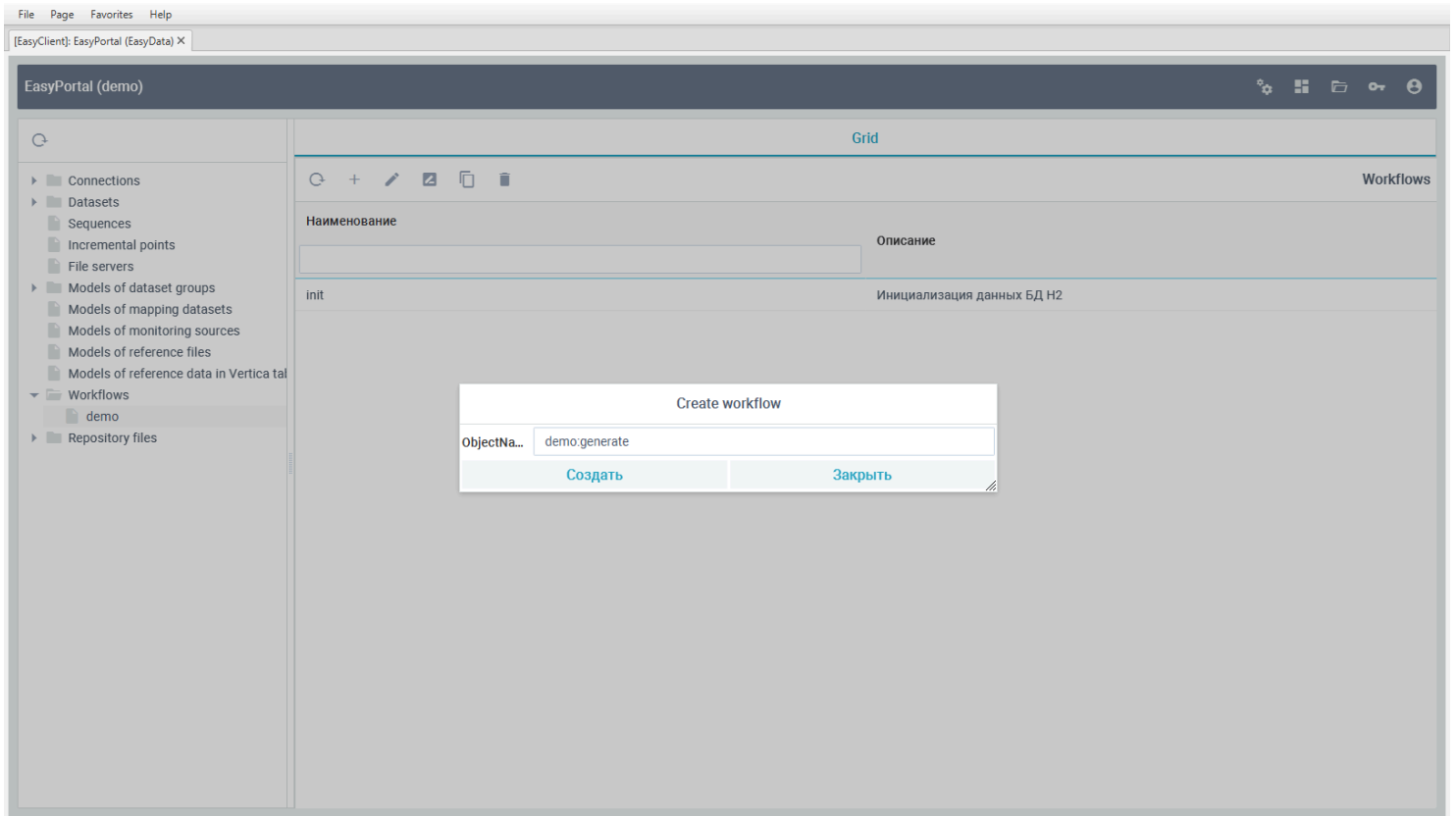
После завершения работы сохраните и закройте модель. Она появится в списке моделей репозитория:



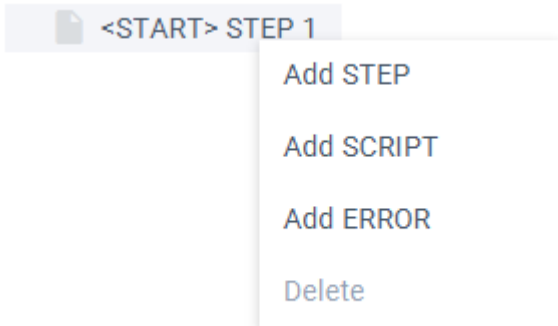


## Создание задачи генерации данных в H2 таблицы

Создайте новую задачу "demo:generate":



Переключитесь в редакторе созданной задачи во вкладку "Процессы" и добавьте в главный шаг "STEP 1" новый скрипт:



Встаньте на добавленный скрипт, нажмите на кнопку выбора шаблона и выберите шаблон "GenerateRandomData":

LibrarieClasses

- ▼ EASYLOADER
  - ▼ ru
    - ▼ easydata
      - ▶ files
      - ▼ generator
        - GenerateRandomData
        - ▶ rdbms
        - ▶ vertica
      - ▶ TFM

## Шаблон GenerateRandomData

Генерирует случайные данные в таблицы по заданным правилам. Рекомендуется использовать для заполнения таблиц или файлов данными для разработки и тестирования.

### Параметры шаблона

- model**: имя модели списка таблиц приёмника для копирования полученных с файлов записей, обязательный параметр;
- included\_list**: шаблоны имен датасетов, включающие имена для обработки датасетов;
- excluded\_list**: шаблоны имен датасетов, исключающие имена для обработки датасетов;
- count\_threads**: количество одновременных потоков обработки данных в модели, значение должно быть больше нуля, по умолчанию 1;
- clear\_tables**: очищать таблицы перед генерацией данных. По умолчанию флаг выключен. Для файловых источников будут удалены файлы датасетов, для jdbc таблиц будет вызван TRUNCATE TABLE (если источник поддерживает этот оператор, в обратном случае будет вызвана очистка записей оператором DELETE);
- bulk\_load**: загружать данные в источник пакетной загрузкой. По умолчанию выключен. Разрешается использовать только, если источник поддерживает такой вид загрузки записей.

### События шаблона

- on\_process\_row**: код обработки сгенерированных записей для таблиц приёмника. Позволяет дополнительно перед записью данных в источник после генерации записи заполнить её поля.

Формат события on\_process\_row

Выбрать
Закреть

Добавьте в параметр "model" созданную модель "random.h2:demo" с помощью ввода имени в текстовое поле или выбрав ее с помощью кнопок вызова редактора параметров с тремя точками:

The screenshot shows the EasyPortal interface with the 'demo:generate' window open. The 'Script 1' configuration is visible, showing the 'model' parameter set to 'random.h2:demo'. The interface includes a left sidebar with a tree view of connections, datasets, and workflows, and a right panel for script configuration.

Имя	Значение	По умолчанию
bulk_load		false
clear_tables	true	false
count_threads	2	1
excluded_list		
included_list		
model	random.h2:demo	

#### Расшифровка параметров:

- clear\_tables** - очистить таблицы перед заливкой в них сгенерированных данных;
- count\_threads** - генерировать в 2 таблицы модели одновременно (многопоточная работа);
- model** - имя модели трансформации со списком таблиц для генерации данных и правилами их заполнения.

Для проверки работы задачи и генерации данных, переключитесь в задаче во вкладку "Отладка" и нажмите "Запуск":

The screenshot shows the 'EasyPortal (demo)' interface with the 'Отладка' (Debug) tab selected. The workflow 'demo:generate' is running, and the log shows the following steps:

Line	Time	Level	Message
13	2022-04-20 22:40:17	FINEST	Execute script "SCRIPT 1" by class ru.easydata.generator.GenerateRandomData with step "STEP 1"...
14	2022-04-20 22:40:18	FINE	*** EasyLoader: генератор записей
15	2022-04-20 22:40:18	FINE	модель: random.h2.demo
16	2022-04-20 22:40:18	FINE	количество таблиц модели: 2
17	2022-04-20 22:40:18	FINE	количество потоков генерации записей в таблицы: 2
18	2022-04-20 22:40:18	FINE	пакетная загрузка в приёмник: Нет
19	2022-04-20 22:40:18	WARNING	Внимание, перед генерацией данных таблицы модели будут очищены!
20	2022-04-20 22:40:18	FINEST	h2.demo.dim: очистка таблицы "DEMO.DIM" ...
21	2022-04-20 22:40:18	FINEST	h2.demo.fact: очистка таблицы "DEMO.FACT" ...
22	2022-04-20 22:40:18	INFO	h2.demo.dim: записано 100 новых записей
23	2022-04-20 22:40:18	FINER	h2.demo.dim: генерация 100 записей для "DEMO.DIM": time 0.345 seconds, 100 rows, 289 rows per second, 0.003450 seconds per row
24	2022-04-20 22:40:18	INFO	h2.demo.fact: записано 1 000 новых записей
25	2022-04-20 22:40:18	FINER	h2.demo.fact: генерация 1 000 записей для "DEMO.FACT": time 0.559 seconds, 1 000 rows, 1 788 rows per second, 0.000559 seconds per row
26	2022-04-20 22:40:19	INFO	--- Execution 1 steps from workflow "##demo_generate##" model completed successfully
27	2022-04-20 22:40:19	INFO	### Workflow demo:generate completed successfully.
28	2022-04-20 22:40:19	INFO	### Job stop
29			Workflow finish

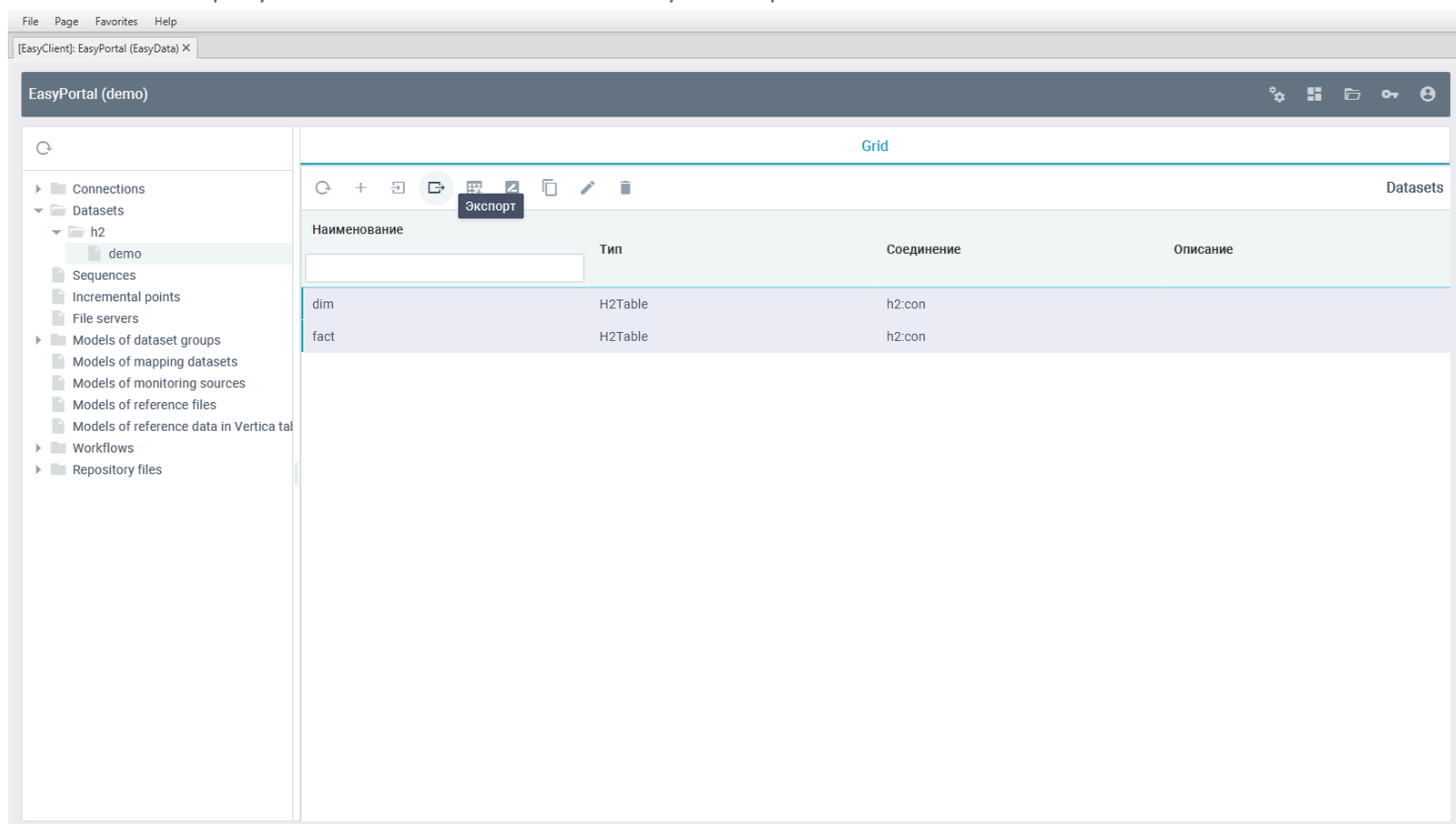
После сохранения и закрытия редактора задачи, она станет доступной в списке задач:

The screenshot shows the 'EasyPortal (demo)' interface with the 'Workflows' tab selected. The workflow list is as follows:

Наименование	Описание
generate	Генерация случайных данных в H2 таблицы схемы DEMO
init	Инициализация данных БД H2

## Экспорт структуры H2 таблиц в файловый источник CSV

Для того, чтобы структуры источника данных скопировать в приёмник данных требуется выполнить экспорт описания датасетов репозитория. Для этого в дереве объектов выберите нужную группу датасетов в "Datasets", пометьте в ней требуемые объекты и нажмите кнопку "Экспорт":



Введите имя группы назначения "csv.demo" и выберите соединение "csv:con":

## Экспорт таблиц

Имя группы назначения \*

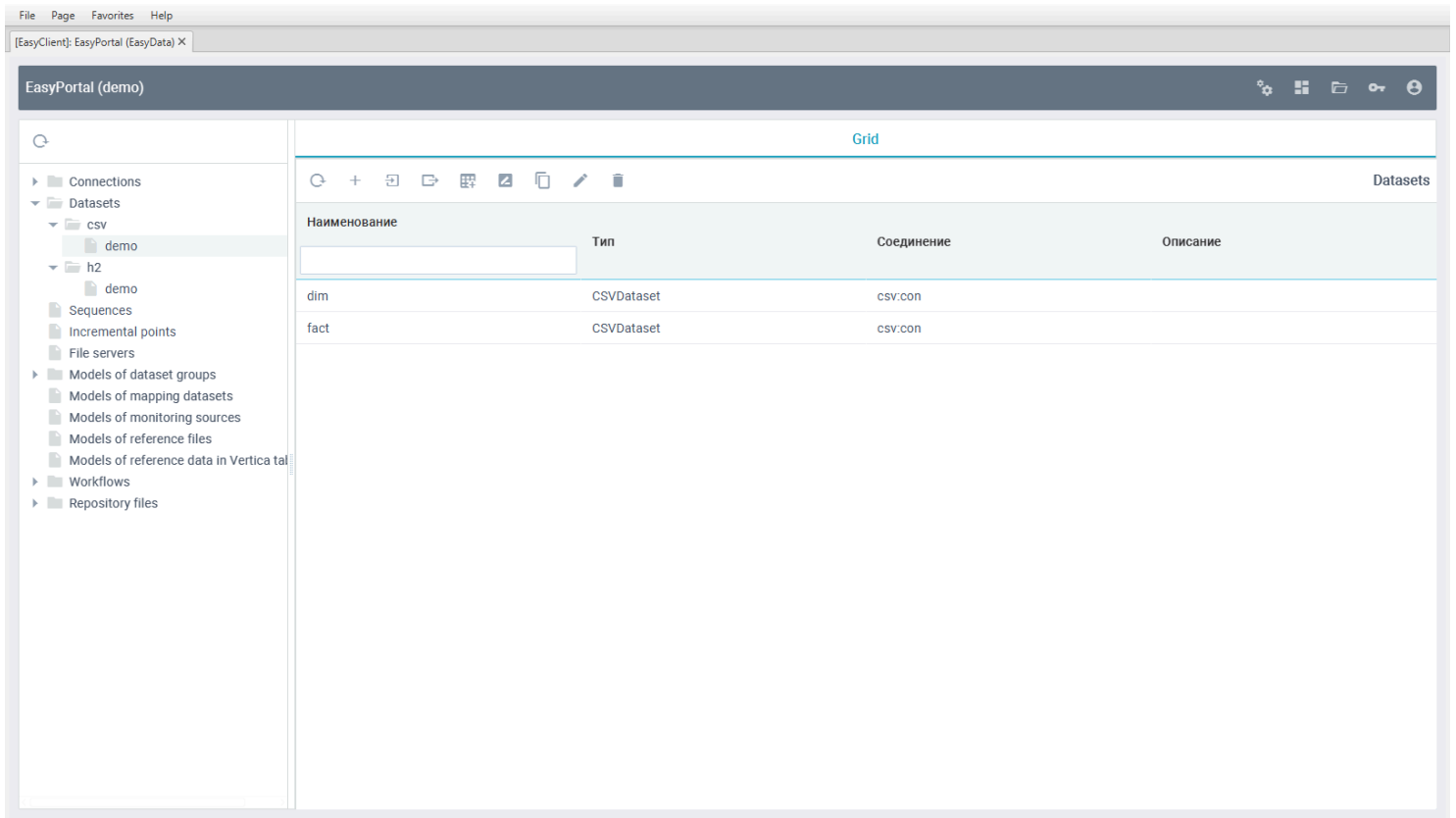
Создать таблицы в приёмнике

csv h2	Имя	Тип
	con	CSVConnection

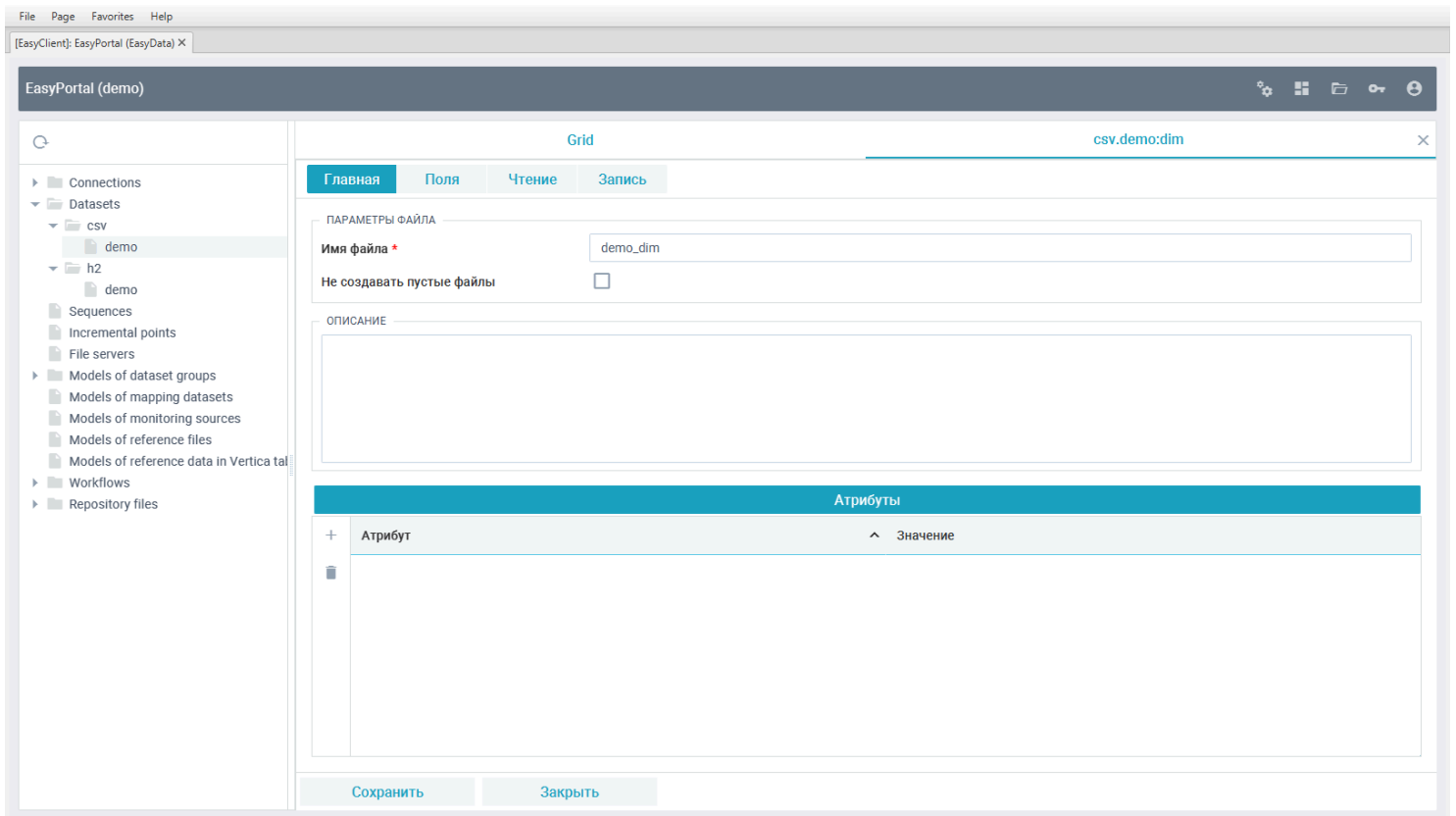
Экспорт

Закреть

После экспорта в группе "csv.demo" появятся новые датасеты для выбранного соединения приёмника данных:



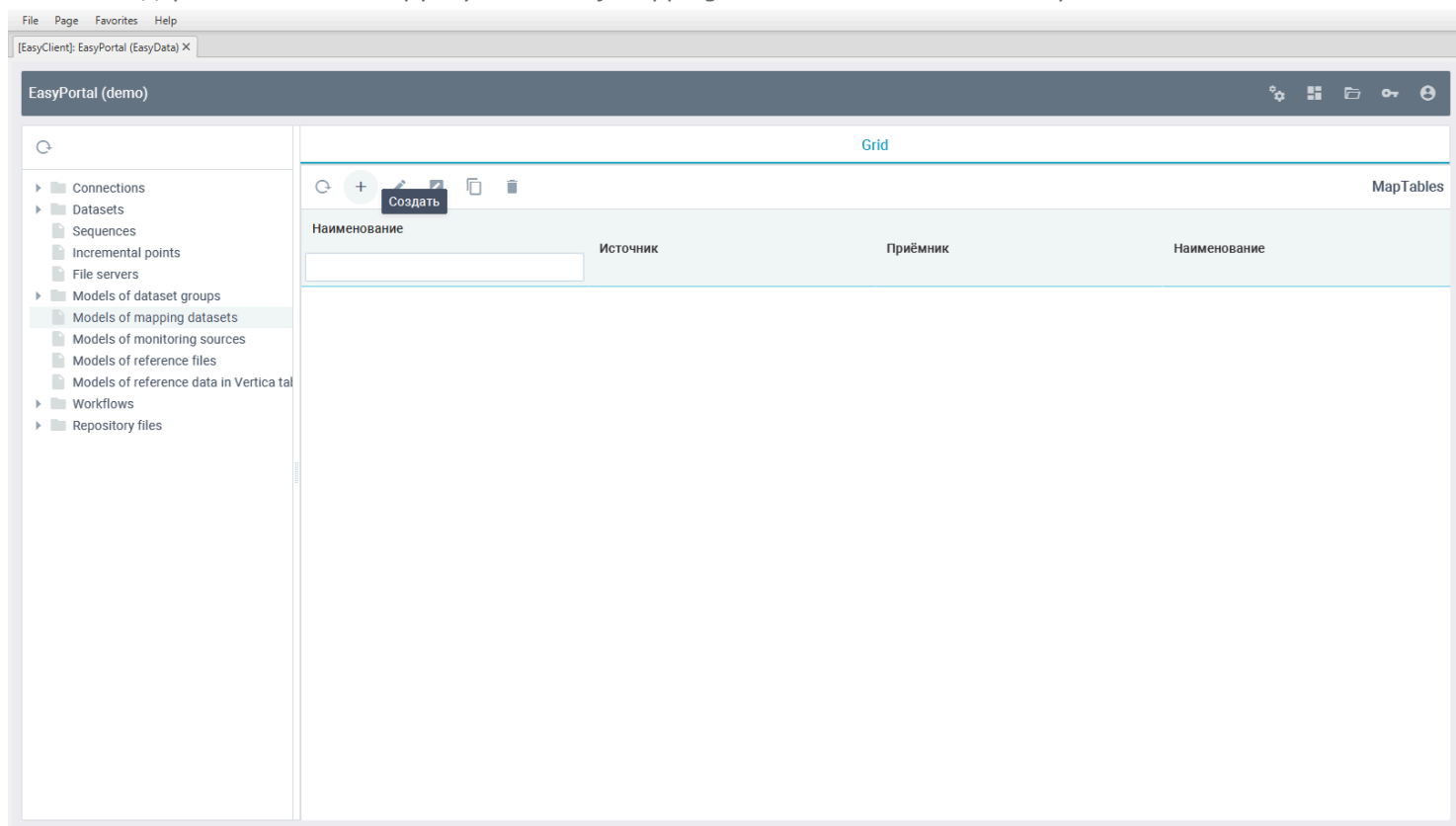
При экспорте структур данных в файловые источники рекомендуется изменить имя файла с генерируемого по умолчанию. Для этого откройте для редактирования "csv.demo:dim" и поменяйте имя файла на "demo\_dim":



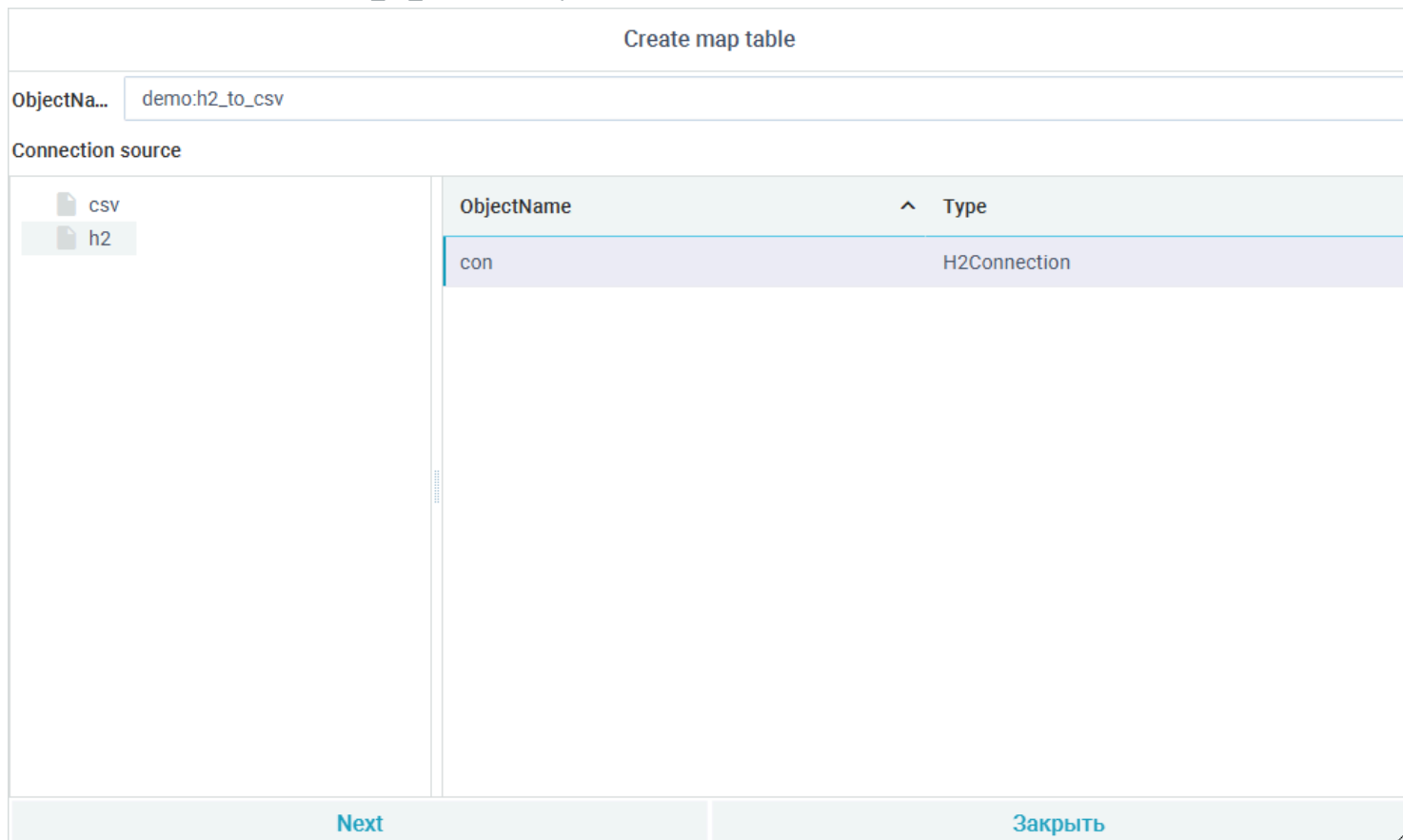
Сохраните изменения и аналогично поменяйте в датасете "csv.demo:fact" имя файла на "demo\_fact".

## Создание модели маппинга данных H2 таблиц и CSV файлов

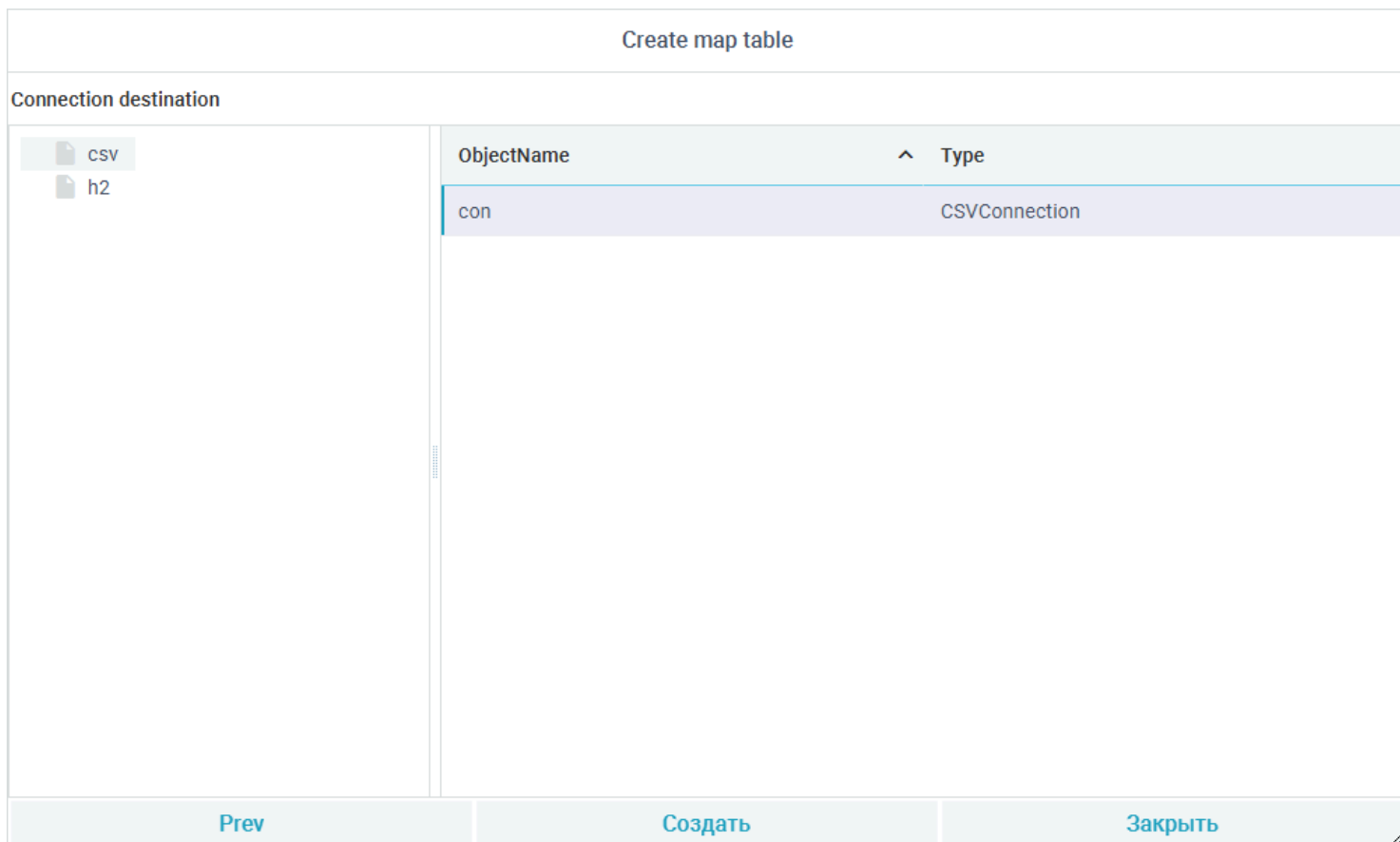
Для копирования данных с источника в приёмник требуется описать правила копирования в модели маппинга. Встаньте в дереве объектов на группу "Models of mapping datasets" и нажмите кнопку "Создать":



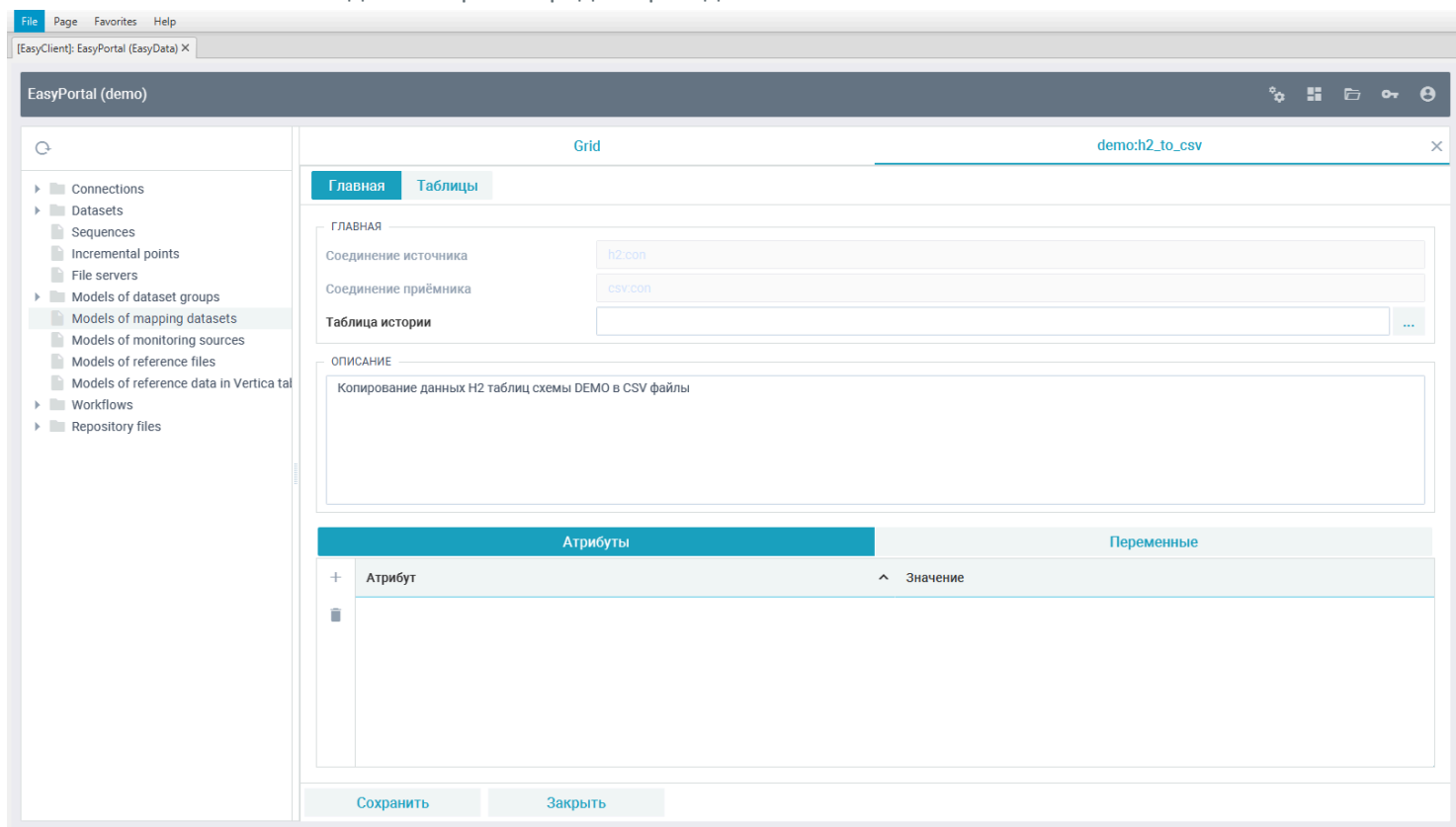
Введите имя модели "demo:h2\_to\_csv" и выберите источником модели соединение "h2:con":



Нажмите кнопку "Далее" и выберите приёмником модели соединение "csv:con":

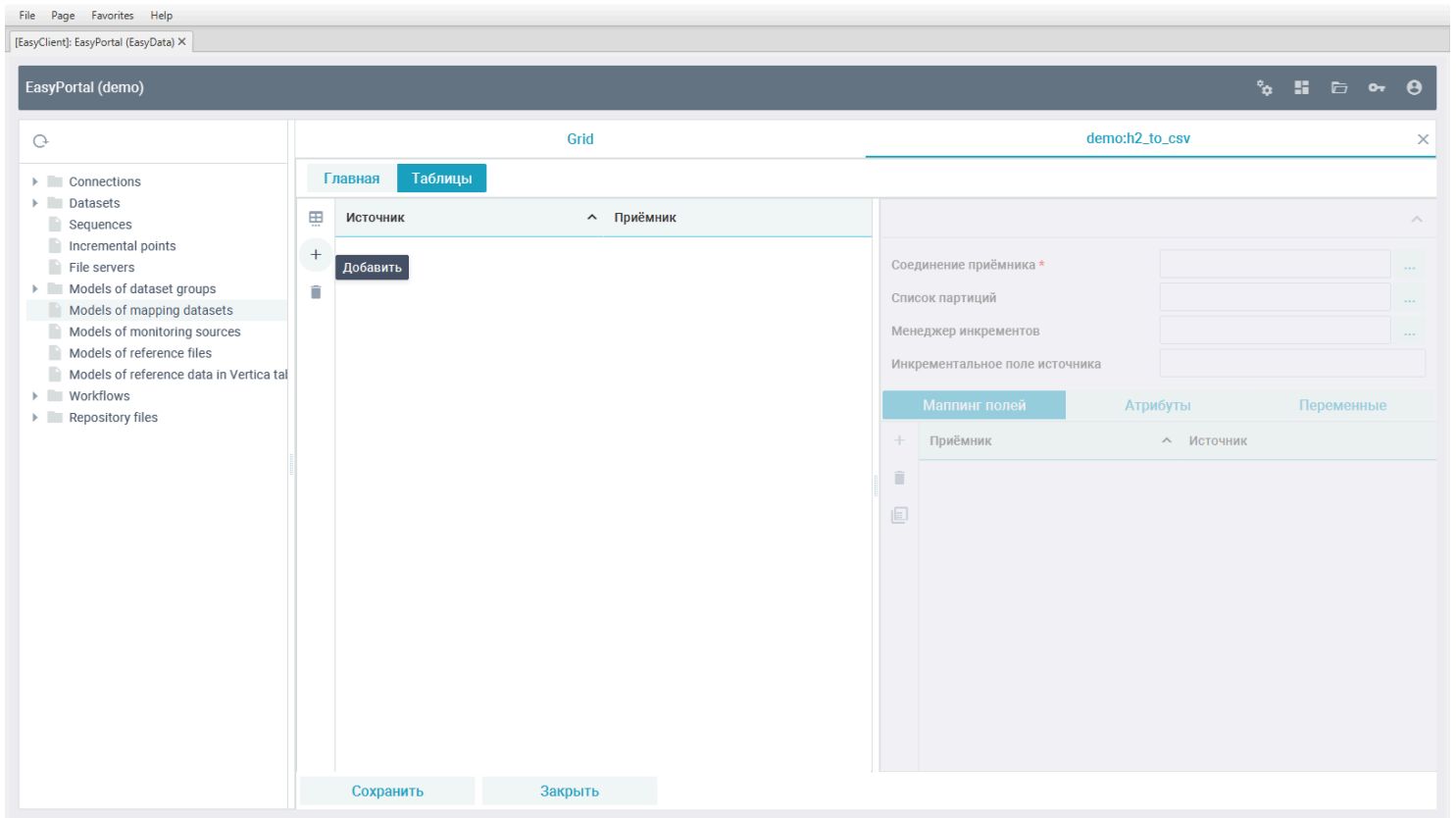


После нажатия кнопки "Создать" откроется редактор моделей:

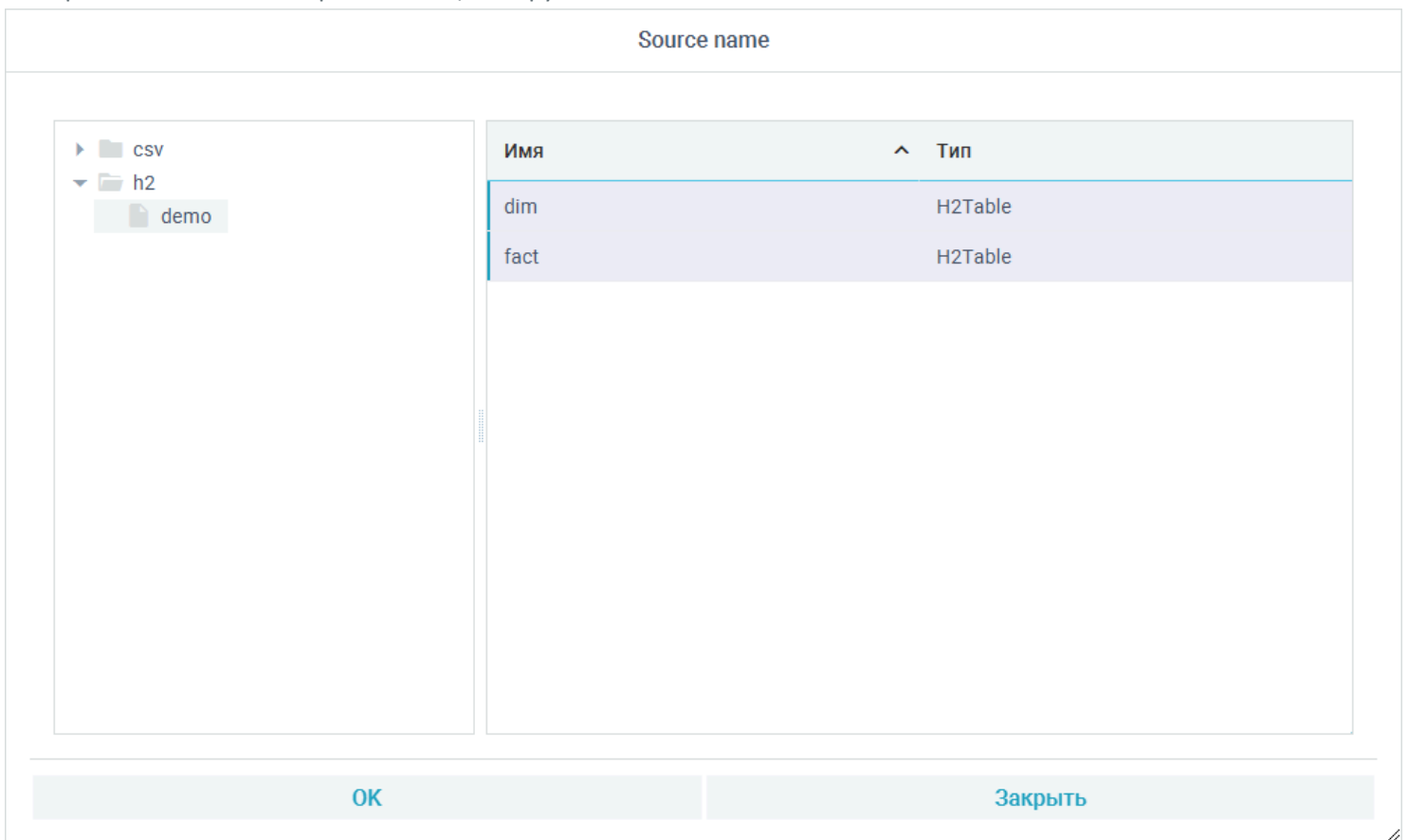


Переключитесь в вкладку "Таблицы" и нажмите кнопку "Добавить":

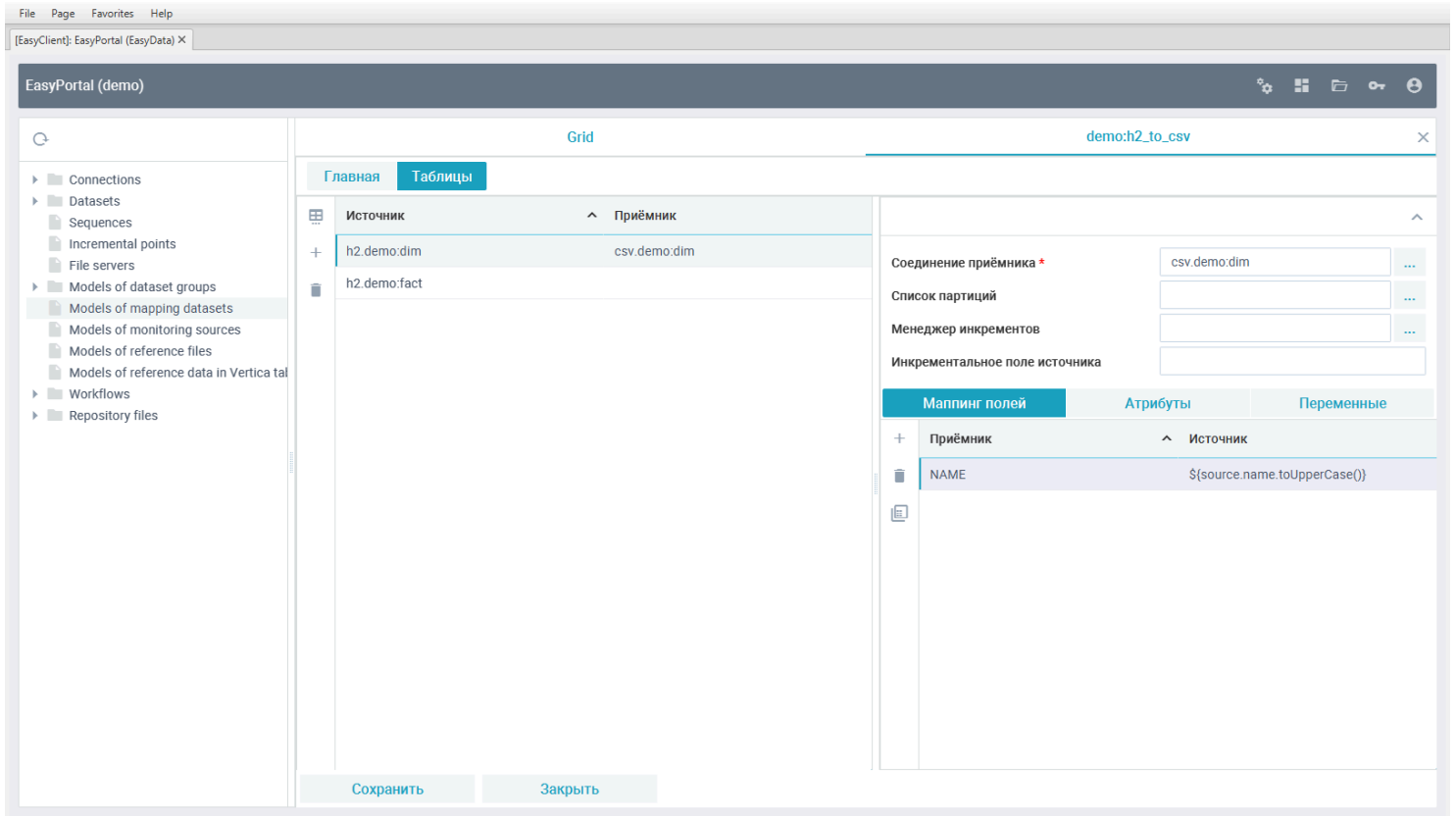




В открывшемся окне выберите таблицы из группы "h2.demo":



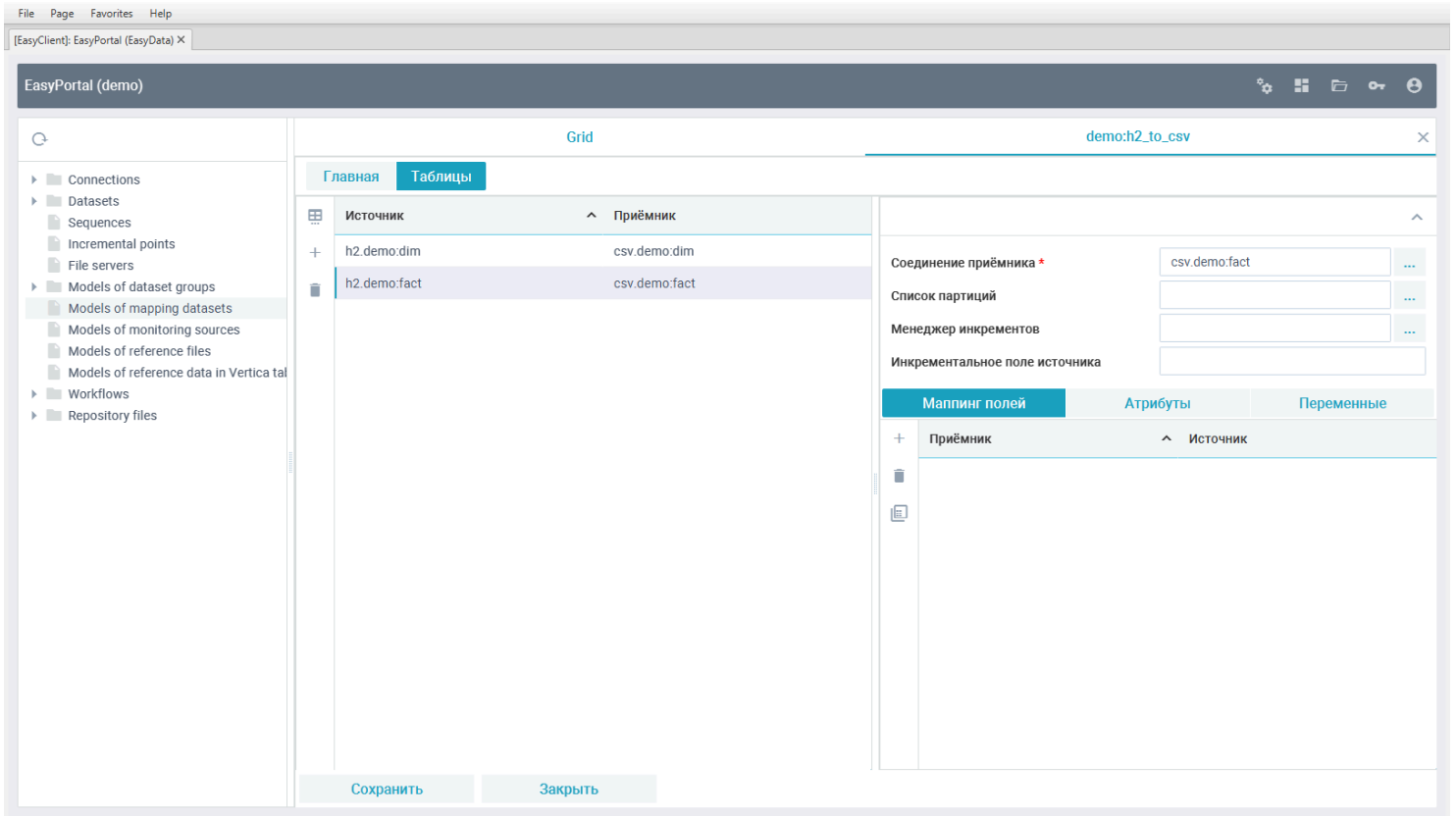
Встаньте на таблицу модели "h2.demo:dim" и укажите в качестве приёмника "csv.demo:dim" в текстовом поле или выбрав с помощью кнопки с тремя точками. В маппинге полей добавьте поле "NAME" и поставьте в источник выражение преобразования "{\$source.name.toUpperCase()}":



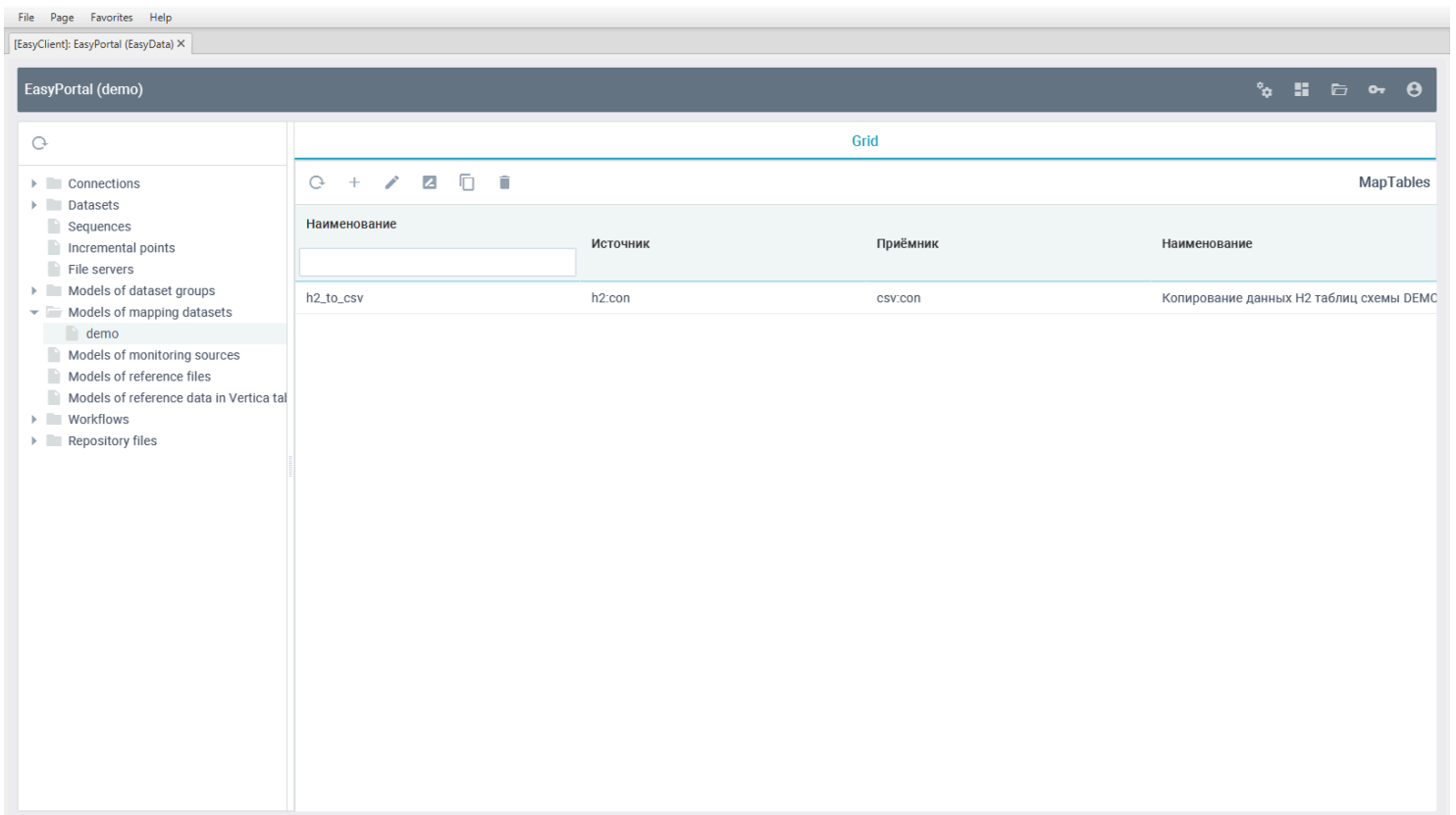
#### Дополнение:

- Значения в маппинге можно вводить как в гриде, так и вызывать с помощью кнопки редактирования более функциональный редактор маппинга;
- Если в маппинге полей модели не указываются поля приёмника, которые по именам присутствуют в источнике, то они автоматически связываются во время выполнения копирования данных;
- Если при копировании данных требуется преобразовать поле источника, то вместо имени поля источника описывается выражение как "*`\${выражение Groovy}`*", где можно обращаться к полям источника в форме "*source.ИмяПоля*".

Переключитесь на таблицу модели "h2.demo:fact" и поставьте ей приёмником датасет "csv.demo:fact":

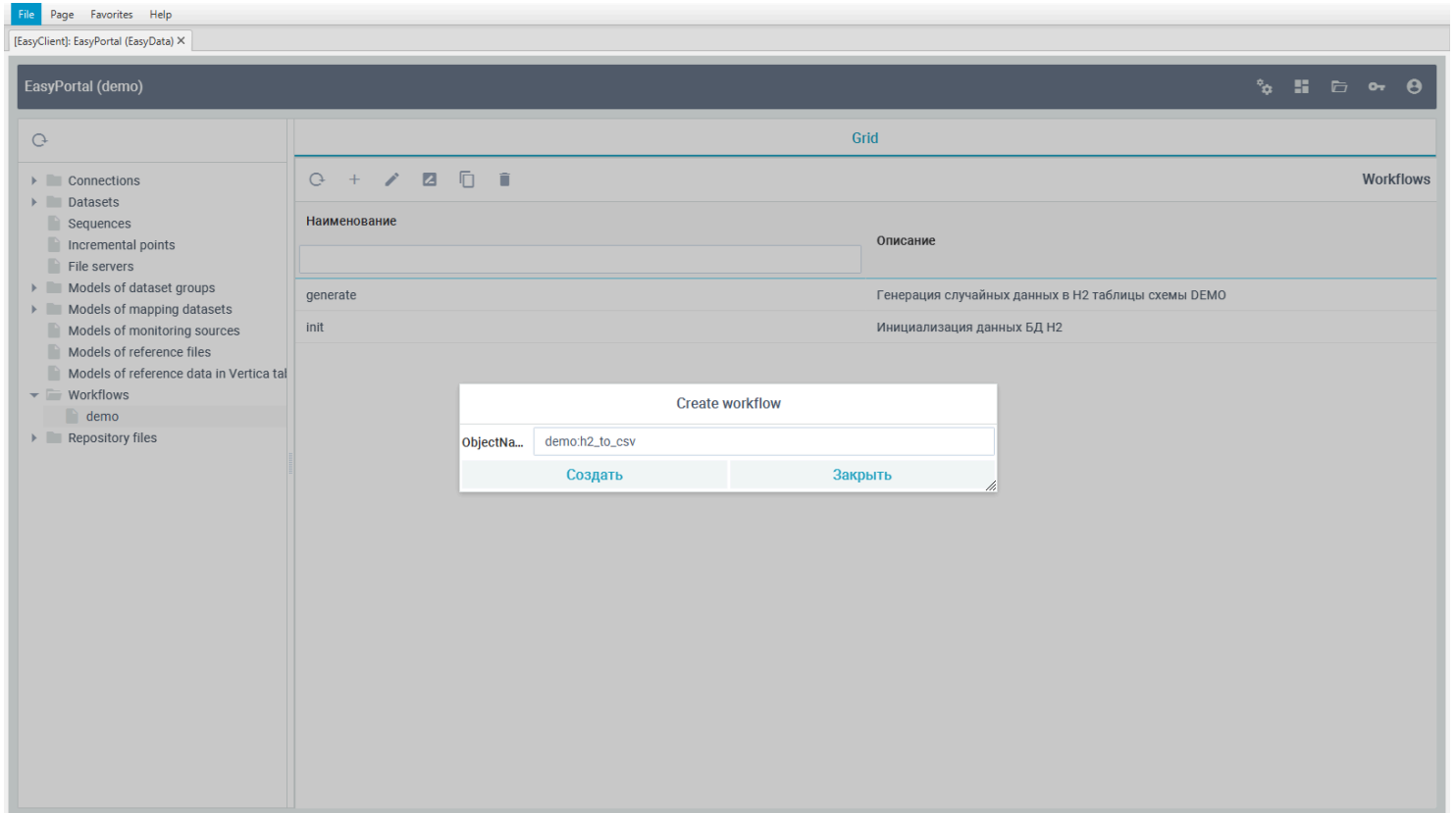


После сохранения модели она появится в списке:

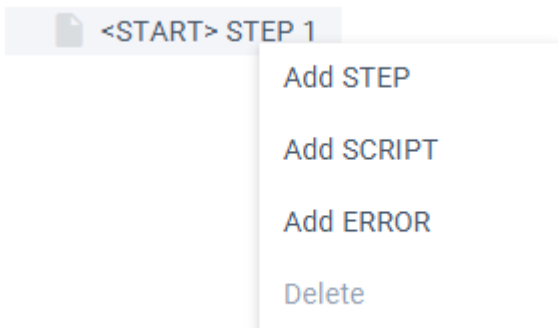


## Создание задачи копирования данных с таблиц в CSV файлы

Создайте новую задачу "demo:h2\_to\_csv":



Переключитесь в редакторе созданной задачи во вкладку "Процессы" и добавьте в главный шаг "STEP 1" новый скрипт:



Встаньте на добавленный скрипт, нажмите на кнопку выбора шаблона и выберите шаблон "CopyToCsv":

LibrarieClasses

- ▼ EASYLOADER
  - ▼ ru
    - ▼ easydata
      - ▶ files
      - ▶ generator
      - ▼ rdbms
        - CopyToCsv
        - CopyToTables
        - LoadToBufferTables
        - RunSql
      - ▶ vertica
    - ▶ TFM

## Шаблон CopyToCsv

Выгружает записи из таблиц и запросов базы данных в CSV файлы. Рекомендуется использовать для иницирующей выгрузки данных из РСУБД для пакетной загрузки в хранилище данных.

### Параметры шаблона

- model: имя модели маппинга выгрузки из таблицы БД в CSV файл, обязательный параметр;
- included\_list: шаблоны имен датасетов, включающие имена для обработки датасетов;
- excluded\_list: шаблоны имен датасетов, исключающие имена для обработки датасетов;
- count\_threads: количество одновременных потоков обработки данных в модели, значение должно быть больше нуля, по умолчанию 1;
- partition\_threads: количество одновременных потоков выгрузки данных из партиций таблицы, значение должно быть больше нуля, по умолчанию 1;
- skip\_exists: пропускать выгрузку таблиц или их партиций, если существует CSV файл с нужным именем, по умолчанию выключено;
- skip\_tables\_in\_history: пропускать выгрузку таблиц или их партиций, если в таблице истории уже существует запись с нужными параметрами, по умолчанию выключено.

### Результат выполнения

Возвращается map со следующими параметрами:

- count\_rows: количество выгруженных записей во все файлы.
- имя\_приёмника: количество выгруженных записей в указанный файл.

Выбрать
Закреть

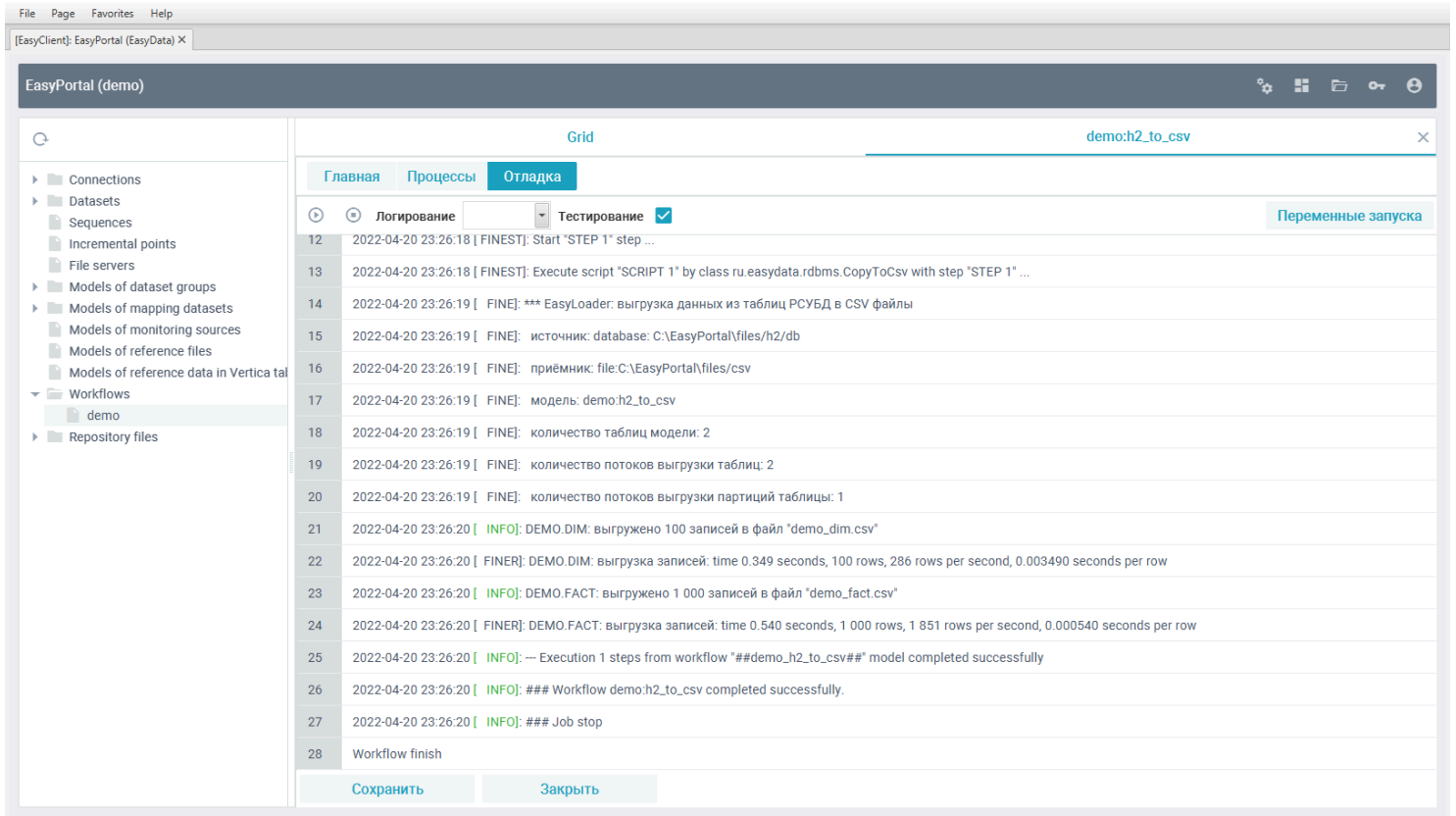
Укажите в параметре скрипта "*model*" имя ранее созданной модели маппинга "*demo:h2\_to\_csv*" и установите количество одновременной обработки таблиц в параметре "*count\_threads*" в 2:

The screenshot shows the EasyPortal interface with the following configuration for a script named "SCRIPT 1":

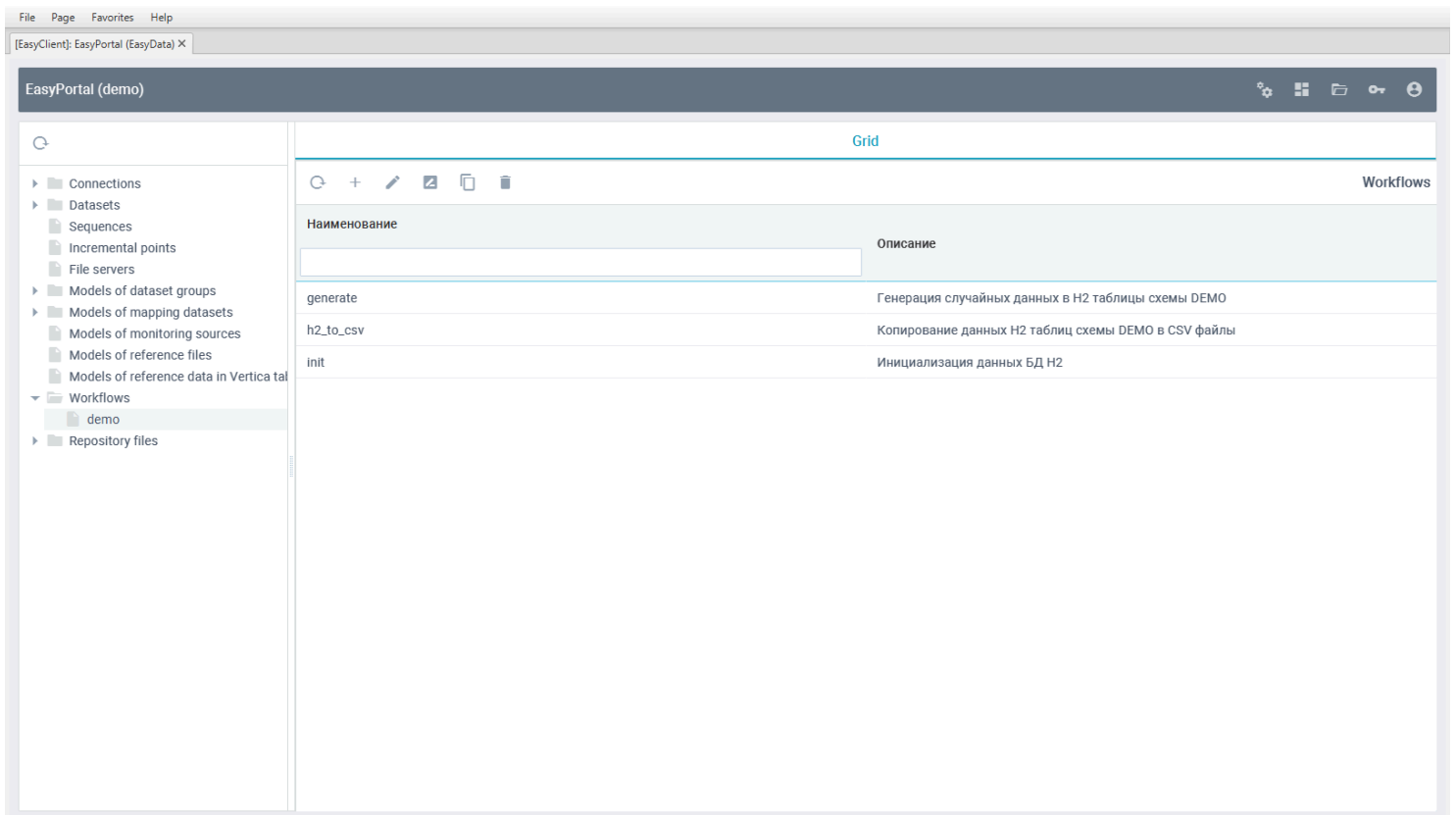
- Template: ru.easydata.rdbms.CopyToCsv
- Script Name: SCRIPT 1

Имя	Значение	По умолчанию
count_threads	2	1
excluded_list		
included_list		
model	demo:h2_to_csv	
partition_threads		1
skip_exists		false
skip_tables_in_history		false

Для проверки работы задачи и копированию данных из таблиц в CSV файлы, переключитесь в задаче во вкладку "Отладка" и нажмите "Запуск":



После сохранения и закрытия задачи она появится в списке задач:



В созданном директории хранения CSV файлов по пути "{DEMO\_PATH}/csv" так же появятся записанные файлы:

```
C:\EasyPortal\files\csv>dir
Том в устройстве C не имеет метки.
Серийный номер тома: 48BC-F319

Содержимое папки C:\EasyPortal\files\csv
20.04.2022  23:26    <DIR>          .
20.04.2022  23:26    <DIR>          ..
20.04.2022  23:26                5 051 demo_dim.csv
20.04.2022  23:26               34 284 demo_fact.csv
           2 файлов                39 335 байт
           2 папок  27 990 822 912 байт свободно

C:\EasyPortal\files\csv>
```

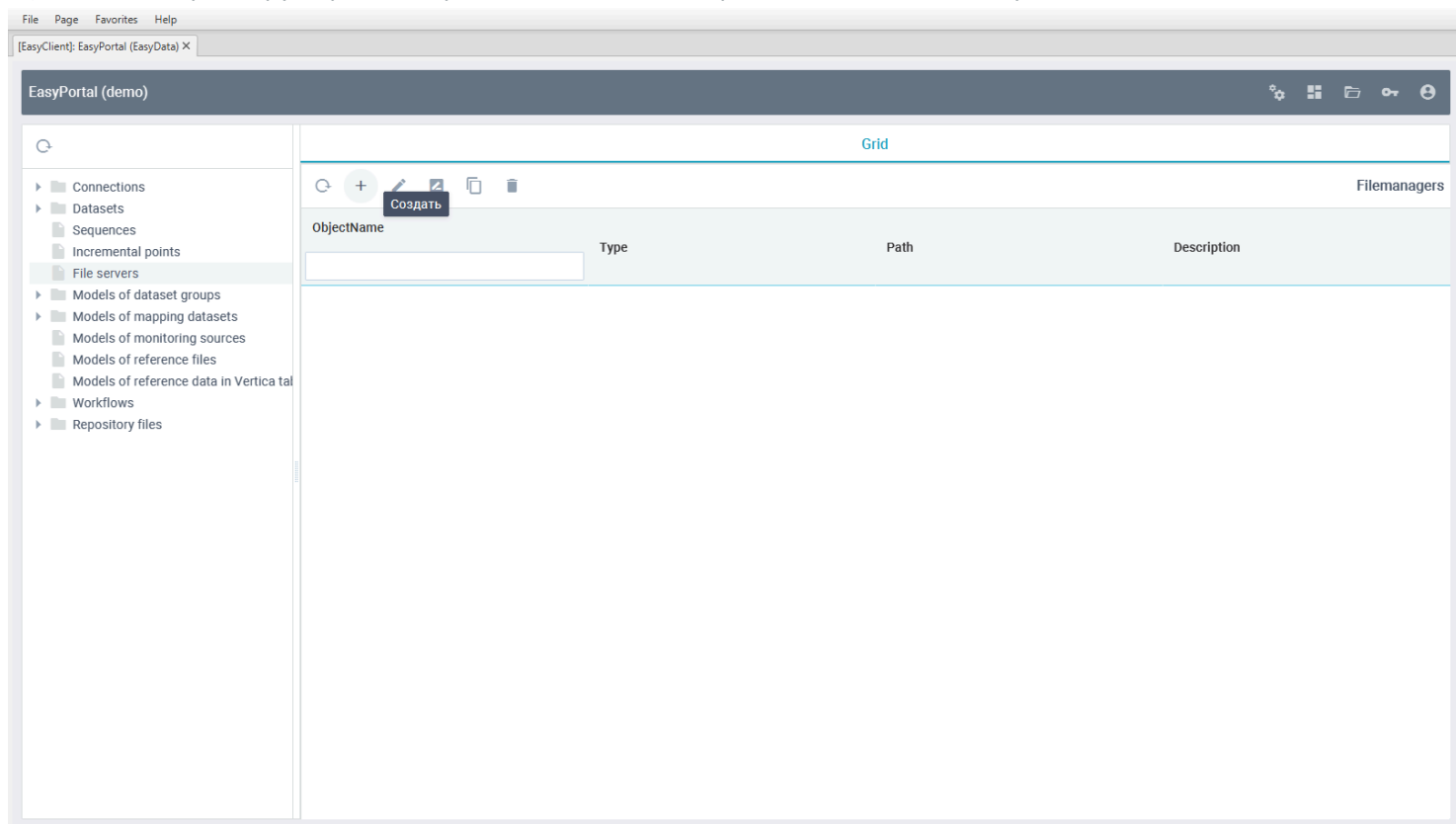
## Генерация эталонных данных

Для работы с примерами данного раздела должны быть выполнены шаги из раздела [Копирование данных из таблиц БД в файлы](#).



## Создание файловых менеджеров эталонных и рабочих CSV файлов

Щелкните в дереве групп репозитория на *File servers* и в гриде нажмите на кнопку *Создать*:



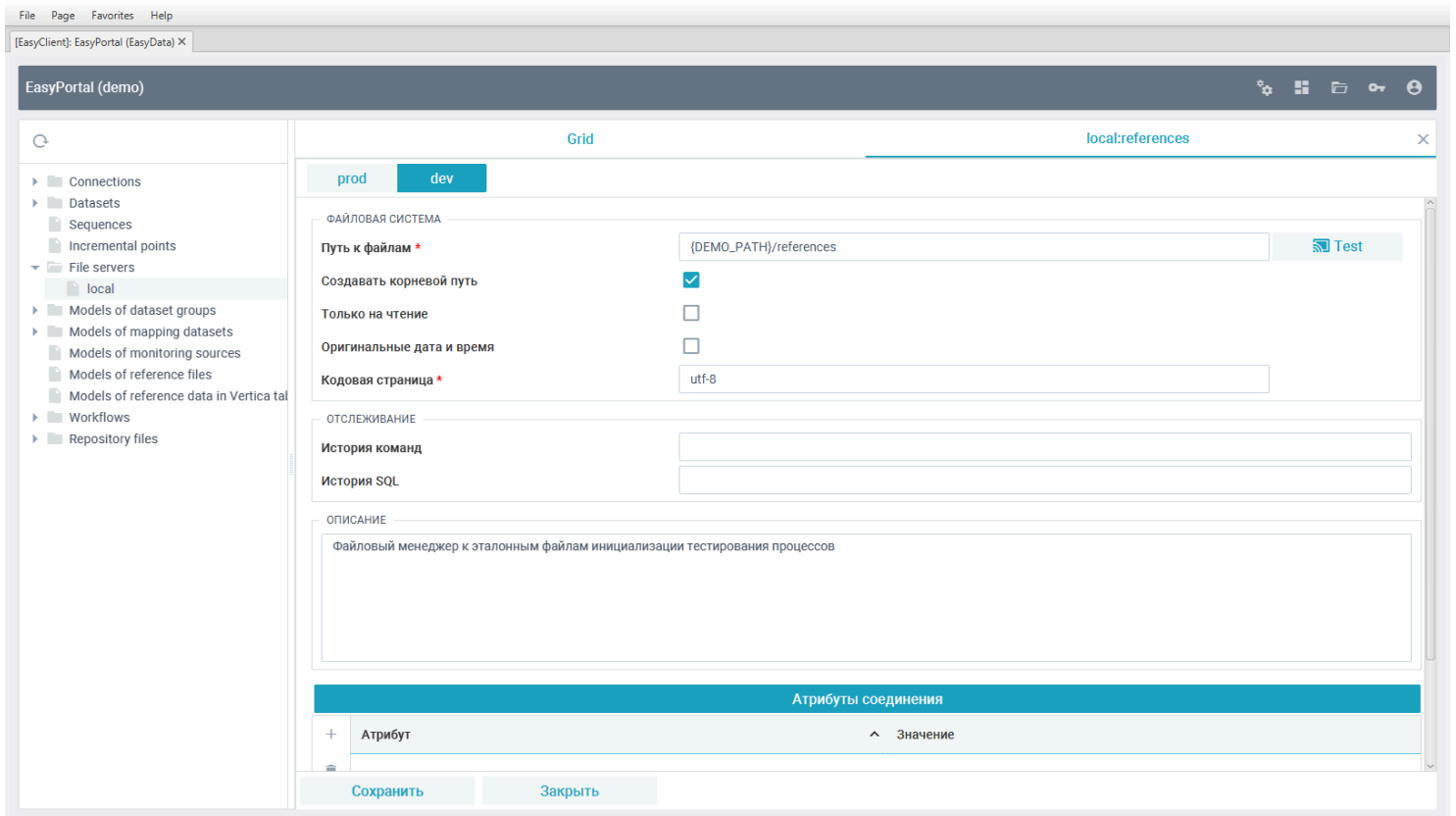
Создайте новый файловый менеджер "*local:references*" с типом "*getl.files.FileManager*":

### Create connect

**ObjectName**

**ClassName**

После нажатия кнопки "*Создать*" откроется редактор файлового менеджера:



На стенде "dev" проделайте следующие действия:

- Вставьте в путь к файлам значение "{DEMO\_PATH}/references". При работе файловый менеджер будет заменять переменную DEMO\_PATH на значение глобальной переменной;
- Включите флаг создания корневого пути.

Создайте новый файловый менеджер "local:csv" с типом "getl.files.FileManager":

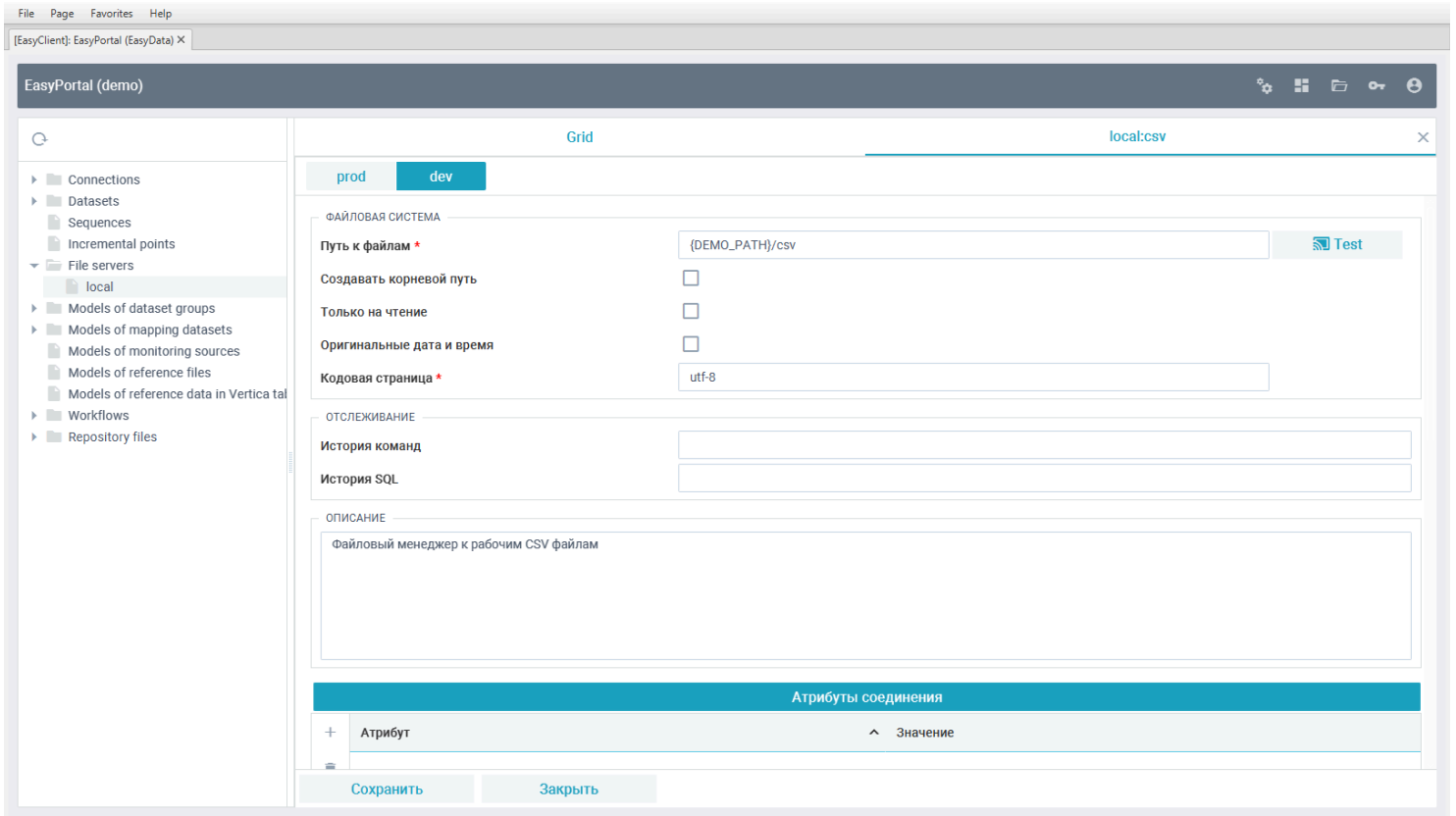
### Create connect

**ObjectName**

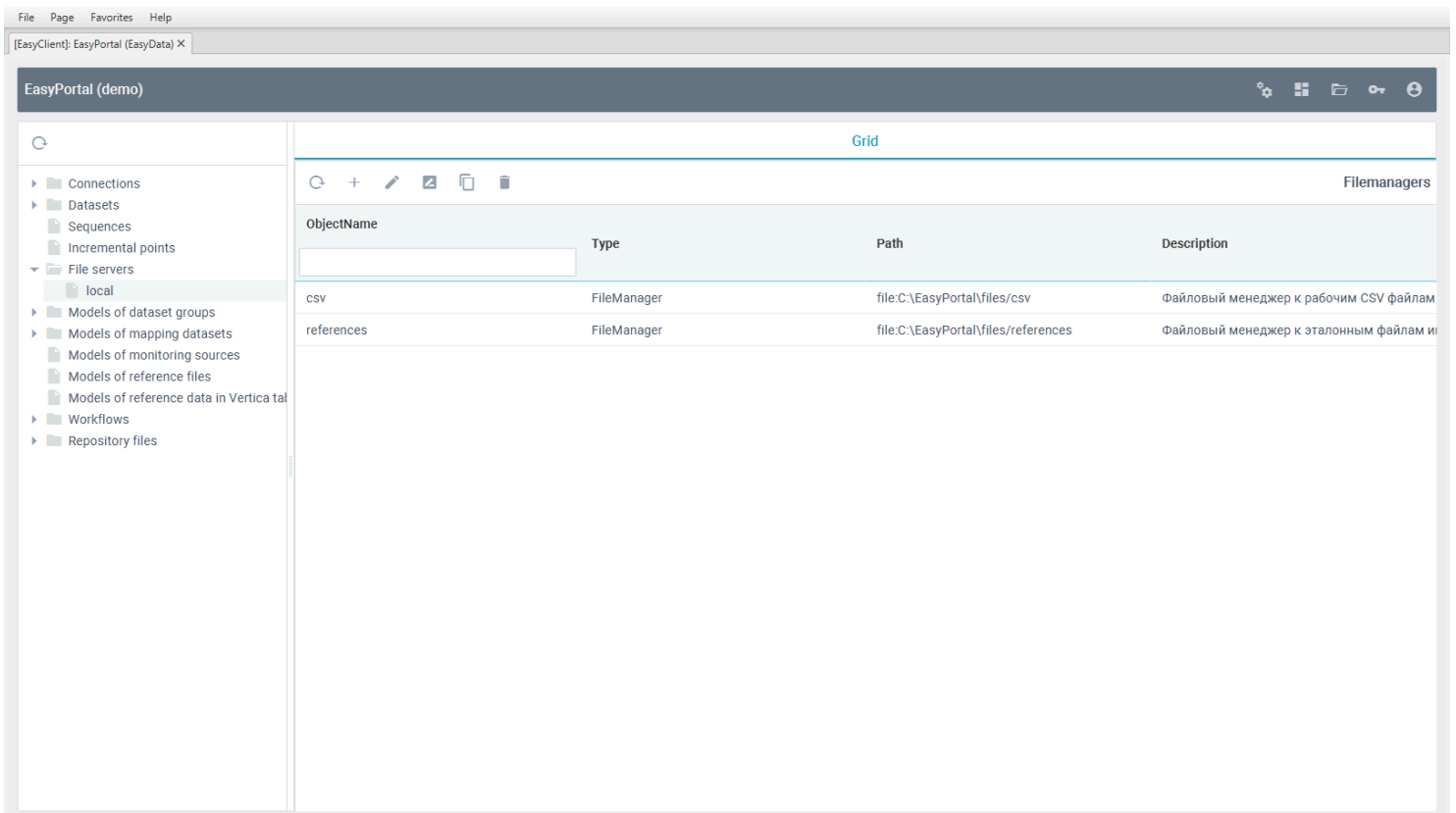
**ClassName**

Создать
Закреть

В открывшемся редакторе файлового менеджера заполните поле пути к файлам значением "{DEMO\_PATH}/csv":

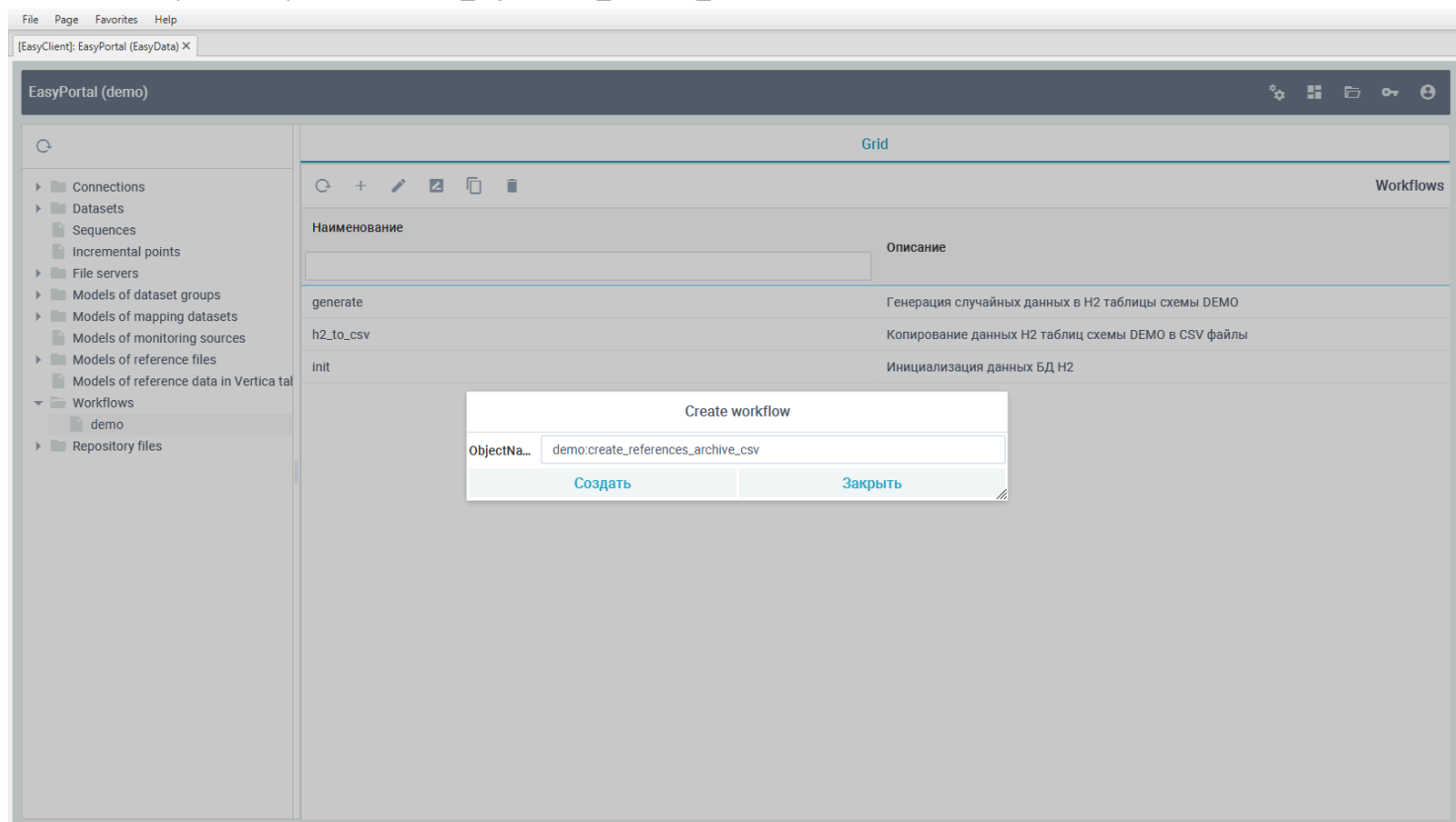


После сохранения файловых менеджеров они появятся в списке:

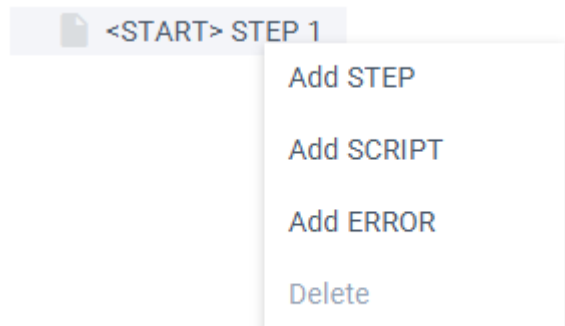


## Создание задачи для генерации архива эталонных CSV файлов

Создайте новую задачу "demo:create\_references\_archive\_csv":



Переключитесь в редакторе созданной задачи во вкладку "Процессы" и добавьте в главный шаг "STEP 1" новый скрипт:



Встаньте на добавленный скрипт, нажмите на кнопку выбора шаблона и выберите шаблон "ArchiveFiles":

LibrarieClasses

- ▼ EASYLOADER
  - ▼ ru
    - ▼ easydata
      - ▼ files
        - ArchiveFiles
        - CleanDirs
        - CleanFiles
        - CopyFiles
        - ProcessDatasetFiles
        - RunProgram
      - generator
      - rdbms
      - utilites
      - vertica

## Шаблон ArchiveFiles

Архивирует файлы по указанной маске с источника, упаковывает в архив и записывает архив в приёмник. Рекомендуется использовать для подготовки архивов для моделей эталонных файлов, для генерации архивов с данными для их доставки в другие системы или отсылки по электронной почте.

### Параметры шаблона

- **source\_manager**: менеджер файлового источника в репозитории, обязательный параметр;
- **source\_path**: относительный путь к файлам источника;
- **source\_file\_mask**: маска поиска файлов на источнике, обязательный параметр;
- **destination\_manager**: менеджер файлового приёмника в репозитории, обязательный параметр;
- **destination\_path**: относительный путь к файлам приёмника;
- **archive\_file\_name**: имя генерируемого архивного файла, обязательный параметр, имя файла указывается без пути;
- **code\_page**: кодовая страница консоли ОС для корректного отображения вывода работы архиватора;
- **in\_memory\_mode**: обработка списка файлов в памяти, рекомендуется выключать при обработке директориев с десятками тысяч файлов для уменьшения потребляемой памяти и включать на малых списках для увеличения скорости работы. Может быть true или false, обязательный параметр, по умолчанию включен;
- **count\_threads**: количество потоков одновременного копирования файлов (по умолчанию 1);
- **directory\_level\_list\_thread**: при чтении списка директориев распараллелить на потоки для указанного уровня вложения директориев;
- **number\_attempts**: количество попыток повтора операции при сбое на источнике или приёмнике, рекомендуется увеличить в случае некачественной связи с источниками.

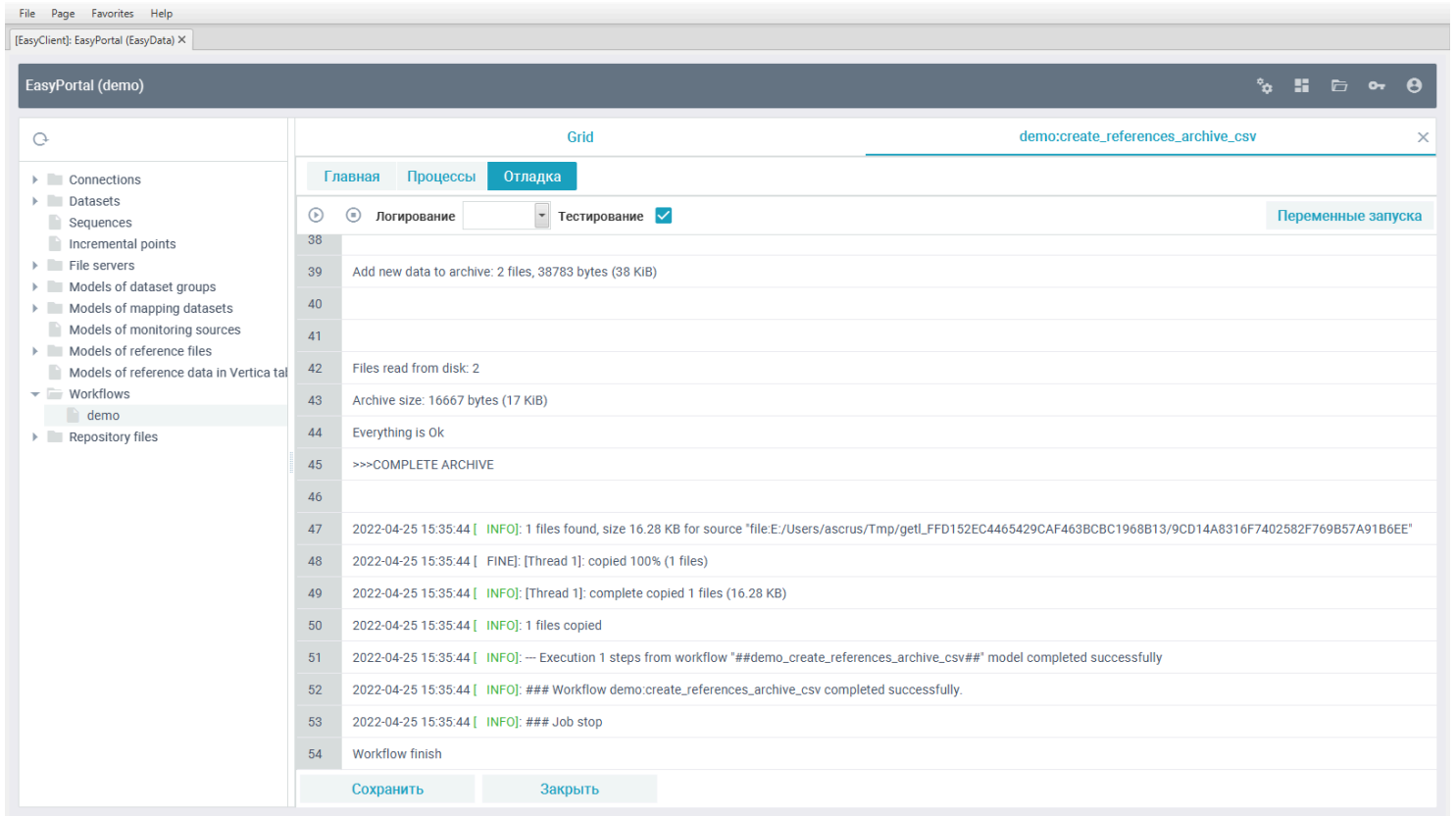
Выбрать
Закреть

Заполните параметры скрипта следующими параметрами:

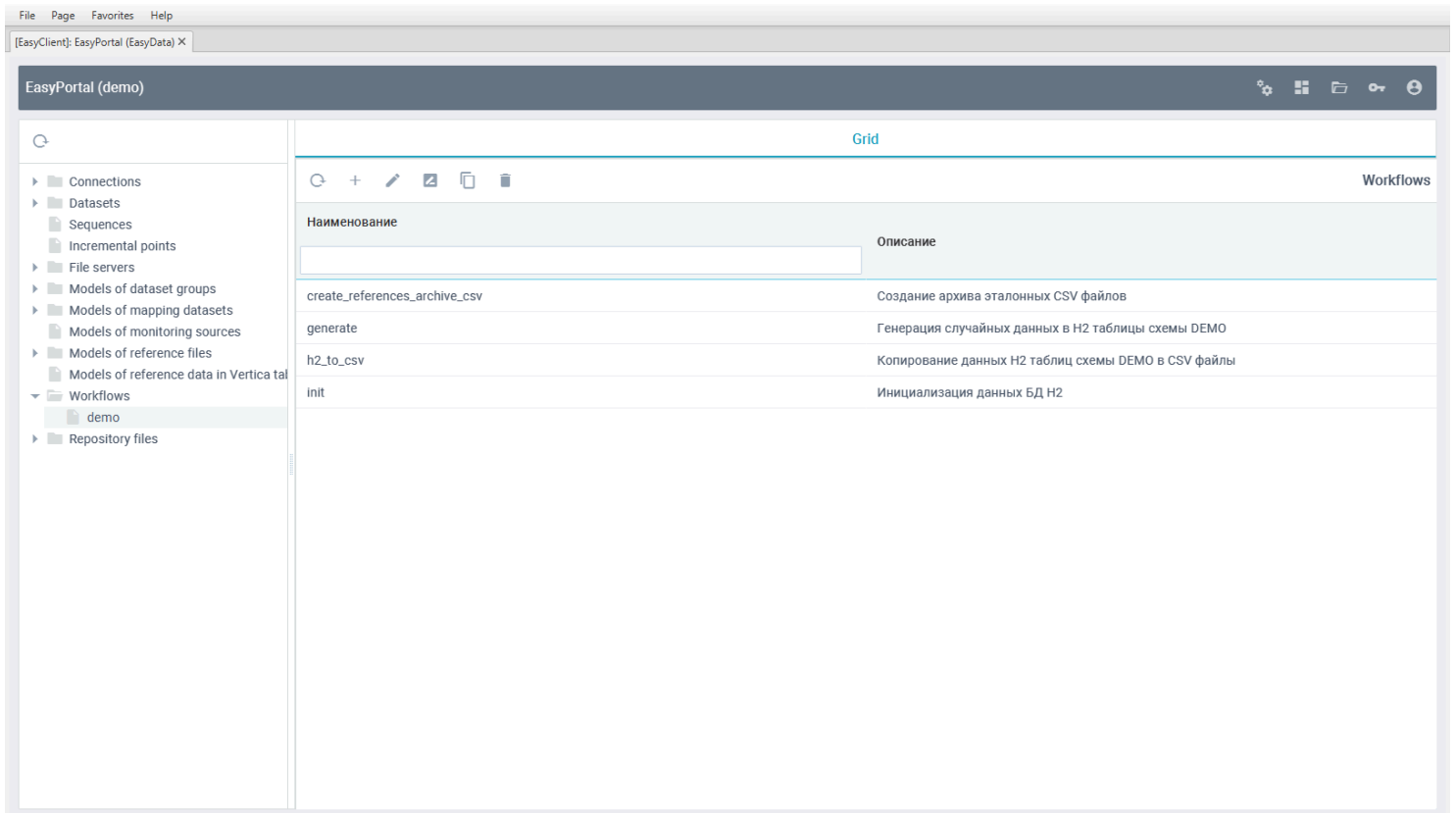
- **archiveFileName**: csv.zip
- **destination\_manager**: local:references
- **source\_file\_mask**: \*.csv
- **source\_manager**: local:csv

Имя	Значение	По умолчанию
code_page		utf-8
count_threads		1
destination_manager	local:references	
destination_path		
directory_level_list_thread		
in_memory_mode		true
limit_count_dirs		
limit_count_files		
limit_size_files		
number_attempts		1
order_files		
source_file_mask	*.csv	
source_manager	local:csv	

Запустите в вкладке "Отладка" задачу, чтобы проверить работу и сгенерировать архив:



После сохранения и закрытия задачи она появится в списке задач:



В пути демо примеров, определенных в переменной "DEMO\_PATH" появится архив "csv.zip":

C:\EasyPortal\files\references\csv.zip

Файл Правка Вид Избранное Сервис Справка

Добавить Извлечь Тестировать Копировать Переместить Удалить Информация

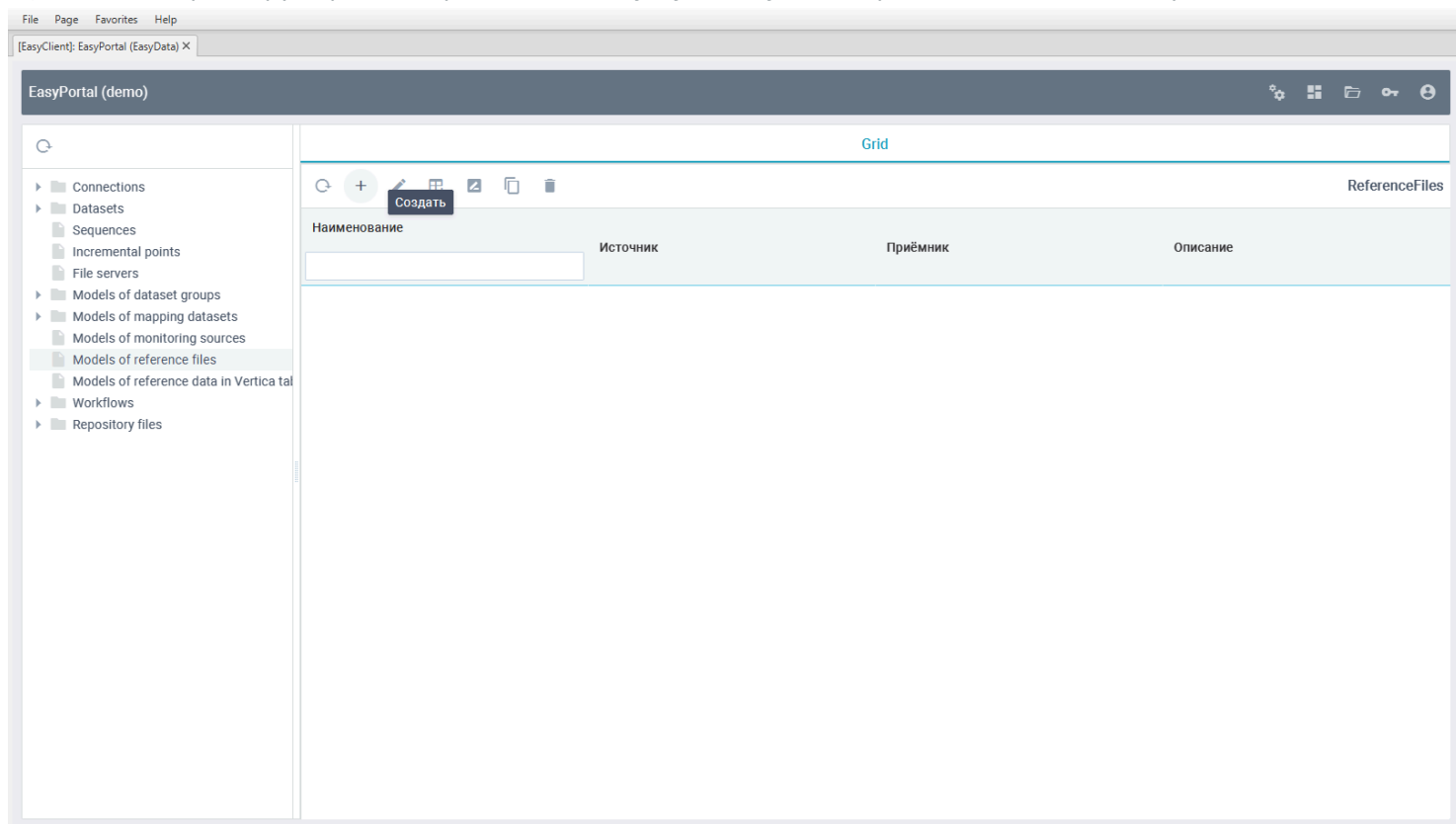
C:\EasyPortal\files\references\csv.zip

Имя	Размер	Сжатый	Изменен	Создан	Открыт	Атрибуты	Зашифрован	Комментарий	CRC	Метод
demo_dim.csv	4 214	2 436	2022-04-20 23:48	2022-04-25 15:35	2022-04-25 15:35	AI	-		B6D8A1D9	Deflat
demo_fact.csv	34 569	13 935	2022-04-20 23:48	2022-04-25 15:35	2022-04-25 15:35	AI	-		6D7E5765	Deflat

Выделено объектов: 0 / 2

## Создание модели эталонных CSV файлов

Щелкните в дереве групп репозитория на *Models of reference files* и в гриде нажмите на кнопку *Создать*:



Введите имя модели "test:csv" и выберите файловый менеджер хранения эталонных файлов "local:references":

**Create reference files**

ObjectNa...

Source manager name

local	ObjectName	Type
	csv	FileManager
	references	FileManager

[Next](#) [Закреть](#)

Нажмите кнопку "Далее" и на следующем экране выберите файловый менеджер файлов приёмника CSV "local:csv":



## Create reference files

Destination manager name

local	ObjectName	Type
	CSV	FileManager
	references	FileManager
<p>Prev Создать Закреть</p>		

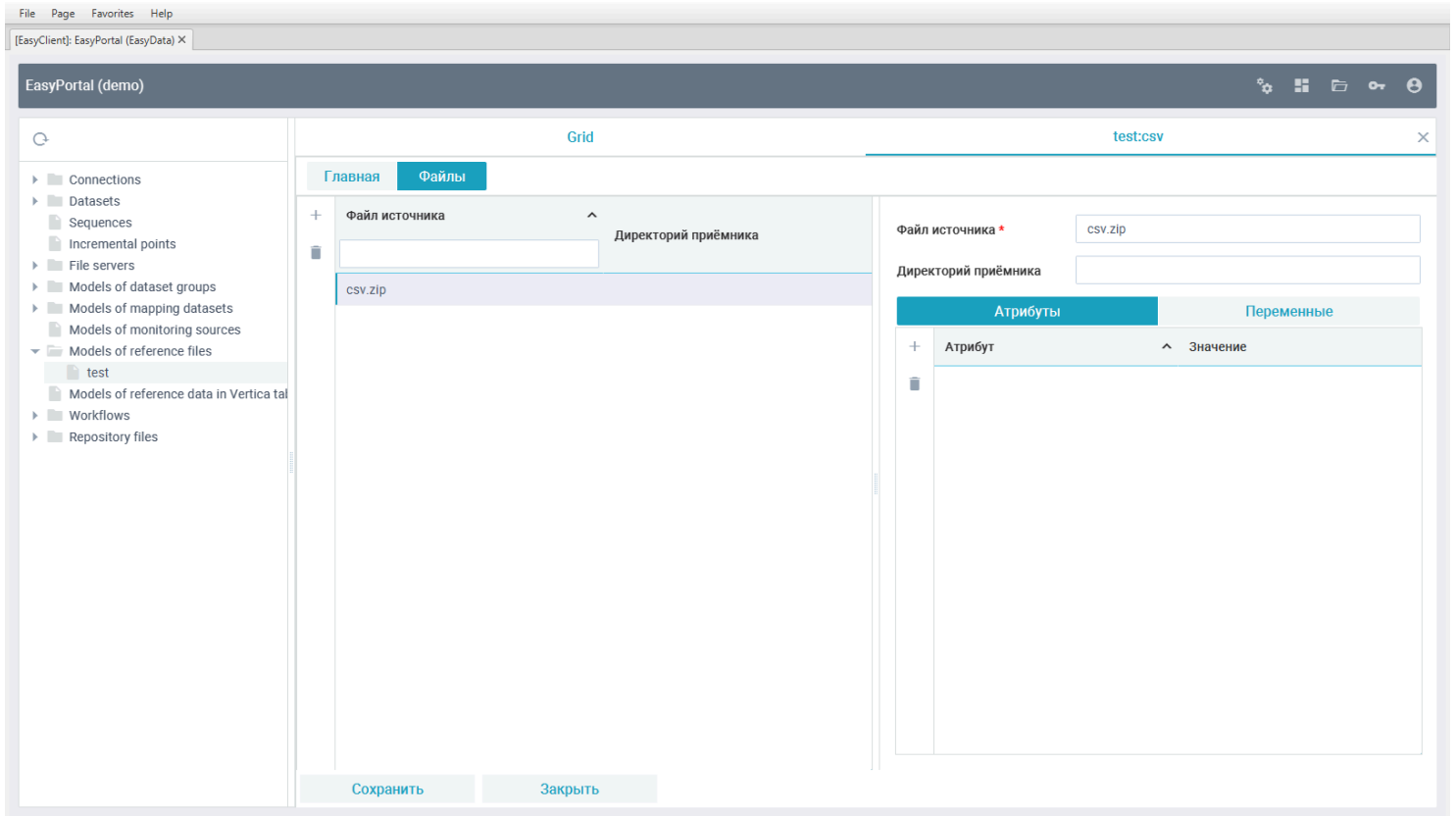
После нажатия кнопки "Создать" откроется редактор модели эталонных файлов. Введите в команду ОС значение "{EASYPORTAL\_HOME}/utils/7z x -y -bd {file}":

The screenshot shows the 'EasyPortal (demo)' interface. The main window is titled 'Grid' and 'test.csv'. The 'Главная' tab is selected, showing the following configuration:

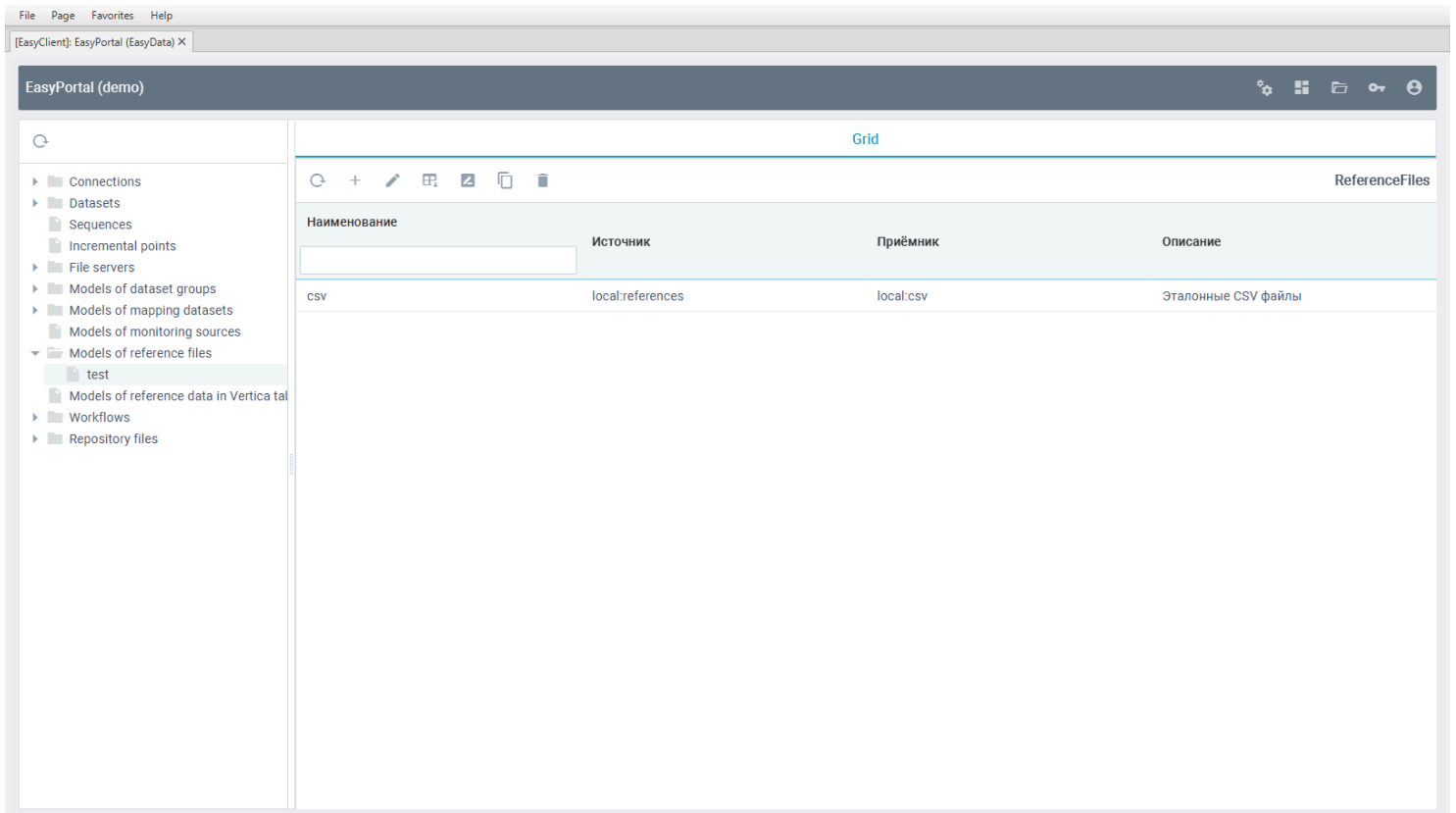
- Источник:** local:references
- Приёмник:** local:csv
- Команда ОС:** {EASYPORTAL\_HOME}/utils/7z x -y -bd {file}
- Создавать директории:**
- ОПИСАНИЕ:** Эталонные CSV файлы

At the bottom, there are tabs for 'Атрибуты' and 'Переменные', and buttons for 'Сохранить' and 'Закреть'.

В вкладке "Файлы" нажмите кнопку "Добавить" и введите в поле "Файл источника" значение "csv.zip":



После сохранения и закрытия модели она появится в списке:



Для проверки разворачивания файлов эталонной модели нажмите на панели кнопку "Заполнить":



Вызвать разворачивание файлов из архива модели в целевую директорию источника

## 1 Вызвать разворачивание файлов из архива модели в целевую директорию источника



Поставьте флаг очистки исходного директория и нажмите "OK":

Fill	
Очистка на источнике	<input checked="" type="checkbox"/>
OK	Закреть

После успешно выполнения в директории "csv" по месту установки демо каталога будут развернуты CSV файлы из архива.