

РУКОВОДСТВО ПО РАЗРАБОТКЕ СКРИПТОВ НА ЯЗЫКЕ C# ДЛЯ ИНСТРУМЕНТА «РЕДАКТОР ЗАДАЧ» ПО PNSoft.

1. Общие данные

Интеграция с ПО ParsecNET (PNSoft) и ParsecOffice (PNOOffice) может осуществляться как посредством уже существующих функций, описанных в документе «Описание сервиса интеграции ParsecNET», так и самостоятельно написанных скриптов на языке C#, использующихся инструментом «Редактор заданий».

Редактор заданий предназначен для создания структуры заданий системе, которые она будет выполнять в соответствии с вашими настройками. За исполнение заданий отвечает специальный сервис системы – "ParsecNET 3 Tasks", который постоянно работает в фоновом режиме как служба Windows.

Выполнение задания может инициироваться:

- Вручную;
- По расписанию;
- По событию устройства.

В свою очередь, инициация по событию устройства имеет следующие атрибуты:

- *Источник события* - указываются компоненты системы, которые порождают события. Данные события инициируют выполнение задания;
- *События* – указывается, какие события будут запускать выполнение задания. Если конкретные события не указаны, то задание будет выполняться по любому событию в системе;
- *Подразделения* – выбираются структурные единицы организации. События, в которых фигурируют субъекты доступа этих подразделений, будут запускать выполнение задания. Данная категория условий запуска задания может использоваться как отдельно, так и совместно с индивидуально указанными в атрибуте *Пользователи* субъектами доступа;
- *Пользователи* – указываются конкретные субъекты доступа. События, в которых фигурируют эти субъекты доступа, будут запускать выполнение задания;
- *Идентификаторы* – выбираются конкретные идентификаторы. События, в которых фигурируют эти идентификаторы, будут запускать выполнение задания;
- *Доп. субъекты* – выбираются субъекты доступа, наличие которых в качестве второго лица при групповом проходе (проходе с сопровождающим) будет запускать выполнение задания. Вместо конкретного субъекта можно выбрать подразделение. В этом случае любой сотрудник данного подразделения при участии «вторым номером» в парном проходе будет инициировать выполнения задания.

Результатом выполнения задания могут быть (но не ограничиваются):

- Команда устройству, входящему в состав СКУД Parsec;
- Отправка текстового сообщения;
- Управление другим заданием;
- Выполнение кода скрипта.

Также возможно задание задержки перед выполнением следующего действия текущего задания.

На данный момент времени существуют следующие поставляемые скрипты для «Редактора заданий»:

- AccessControl_ActivateBlockedCardByOperator - снятие блокировки картой оператора;
- AccessControl_InactiveAccessSubjects - управление неактивными субъектами доступа и идентификаторами;
- AccessNotify_Telegram - отправка уведомления о событии в мессенджер Telegram;
- EmailReportTask - создание отчета УРВ и отправка его на указанный адрес электронной почты;
- ReportToFileTask - создание отчета УРВ и сохранение его в выбранной директории;
- EventReportToEmailTask - создание отчета по событиям системы и отправка его на указанный адрес электронной почты;
- EventReportToFileTask - создание отчета по событиям системы и сохранение его в файл;
- IDIS_MessageTask - отправка сообщения в видеосистему IDIS;
- ITVTelemetryTask - автоматизация работы ИСБ "Интеллект";
- ReportWholsHereToEmailTask - создание отчета "Не покидали территорию" и отправка его на указанный адрес электронной почты;
- ReportWholsHereToFileTask - создание отчета "Не покидали территорию" и сохранение его в файл;
- ShellExecuteTask - запуск внешнего исполняемого файла;
- SmsNotify - отправка SMS сообщения через интернет-портал;
- SmsNotifyOnExtraFieldTelephone - отправка SMS сообщения на телефон, указанный в системном дополнительном поле;
- VisitorControl_InactiveRequests - управление заявками бюро пропусков.

Эти файлы находятся в папке SCRIPTS в каталоге установки ПО Parsec.

Для самостоятельной разработки скриптов для «Редактора заданий» необходимо соблюдение правил и требований, описанных в следующем разделе.

2. Требования к скриптам для модуля «Редактор заданий» ПО PNSoft

- Скрипты разрабатываются на языке C#.
- Файл скрипта рекомендуется сохранять в кодировке UTF-8.
- Код скрипта будет выполняться внутри процесса сервиса "ParsecNET 3 Tasks" ("...\MDO\ParsecNET 3\ParsecTaskClient.exe").
- В файле скрипта должен быть объявлен класс (структура), реализующий интерфейс `MDO.Parsec.ExternalTasks.IExternalTask`:

```
using System;
using System.Windows.Forms;

namespace MDO.Parsec.ExternalTasks
{
    public interface IExternalTask
    {
        /// <summary>
        /// Название скрипта
        /// </summary>
        string name { get; }

        /// <summary>
        /// Описание скрипта
        /// </summary>
        string description { get; }

        /// <summary>
        /// Метод, который будет выполняться при запуске задания
        /// </summary>
        ///
        /// <param name="context">Контекст выполнения скрипта<see cref="ITasksContext"/>
        /// Содержит:
        /// - сведения о машине, на которой выполняется скрипт;
        /// - данные события, которое инициировало запуск скрипта (для случая запуска
        задания по событию);
        /// - набор данных, которые скрипты могут сохранять в памяти службы ParsecNET 3
        Tasks и использовать для хранения между запусками.
        /// </param>
        ///
        /// <param name="parameters">Входные параметры скрипта, заданные с помощью GUI-
        диалога настроек пользователем</param>
        void run(ITasksContext context, string parameters);

        /// <summary>
        /// Есть ли GUI-диалог настроек у скрипта?
        /// </summary>
        bool useParametersDialog { get; }

        /// <summary>
        /// Метод, который вызывает GUI-диалог параметров настройки скрипта
        /// </summary>
        /// <param name="owner">Handler окна, которое будет Parent'ом отображаемого
        диалога</param>
        /// <param name="parameters">Текущий набор параметров скрипта в виде
        строки</param>
        /// <returns>Новый набор параметров скрипта, который пользователь задал с помощью
        GUI</returns>
        string showParametersDialog(IWin32Window owner, string parameters);
    }
}
```

Код интерфейса MD0.Parsec.ExternalTasks.ITaskContext:

```
using System;
using System.Collections.Generic;
using MD0.Parsec;

namespace MD0.Parsec.ExternalTasks
{
    public interface ITaskContext
    {
        /// <summary>
        /// Guid текущей рабочей станции, на которой выполняется код скрипта
        /// </summary>
        Guid localId { get; }

        /// <summary>
        /// Guild сервера
        /// </summary>
        Guid serverId { get; }

        /// <summary>
        /// Событие, вызвавшее запуск задания (будет != null, если задача выполняется по
событию)
        /// </summary>
        EventArgs firedEvent { get; }

        /// <summary>
        /// Dictionary объектов сессии Task-manager'a
        /// Важно! Словарь является общим для всех задач, выполняемых данным экземпляром
сервиса ParsecNET 3 Tasks.
        /// </summary>
        Dictionary<string, object> context { get; }
    }
}
```

- Для получения данных события, инициировавшего запуск скрипта, используйте следующие фрагменты кода:

```
Event evt = new Event(context.firedEvent, new EventParser());

// Guid события (уникальный)
Guid eventID = evt.Data.ID;

// Guid связанного родительского события
Guid parentID = evt.Data.ParentId;

// дата и время события
DateTime eventTime = evt.Data.Date;

// код события
uint eventCode = evt.Data.Code;

// битовая маска категорий события (каждое событие может принадлежать нескольким
категориям сразу),
// описание битовой маски классов и всех типов событий см. во вложенном excel-документе
UInt64 categoriesMask = evt.Data.Mask;

// код карты в 16-чном формате (только для транзакций с пользователем/картой, например
"Нормальный вход по ключу")
string cardCodeHex = evt.GetCardCode();

// Guid субъекта доступа (если он есть в событии) [посетителя, сотрудника] (только для
авторизованных транзакций)
Guid userID = evt.GetUserID();
```

```
// Guid компонента – источника события
Guid componentID = evt.Data.Component;

// Функция получения значения поля данных события по ID,
// список ID полей доступен в таблице 1
object evt.GetFieldValue( Guid fieldID )

// Функция получения строкового представления поля данных события по ID,
// список ID полей доступен в таблице 1
string evt.GetFieldValueString( Guid fieldID )
```

Таблица 1.

ID поля	Описание
71b03d7b-2e11-47cd-bf47-adaf320aeb10	Дата события
c7ad4f51-d8af-4944-bf92-23714715147e	Время события
2c5ee108-28e3-4dcc-8c95-7f3222d8e67f	Дата/время события
633904b5-971b-4751-96a0-92dc03d5f616	Источник события (наименование территории или оператора)
42dab9c6-5d30-4030-8ccd-2cad6fcbcb5f2	Источник события (массив идентификаторов территорий)
d1847aff-11aa-4ef2-aaaa-795ceefe5f9f	Тип события (наименование)
9f7a30e6-c9ed-4e62-83e3-59032a0f8d27	Идентификатор события (Guid)
c4ae9465-8375-4169-ba61-eb7e365a7352	Тип события (код в 16-ричной системе счисления)
57ca38e4-ed6f-4d12-adcb-2faa16f950d7	Тип события (код в 10-чной системе счисления)
68ef9fd3-a72d-4520-9c63-5c37b0ae8539	Субъект (ФИО) – из dictionary
7c6d82a0-c8c8-495b-9728-357807193d23	Идентификатор субъекта (PERS_ID - Guid)
4c5807cb-2c06-4725-9243-747e40c41d6c	Область (название)
2ab38696-1e30-4e04-a956-b951cb7c2033	Деталь (название)
89c9d5ac-6e13-4715-a524-7c3b34931385	Рабочая станция
fea92e1c-e07d-4932-a6a1-e8c53e3087d9	Оператор
03ceb65f-dcad-4b56-94b8-be9fdb463988	Подробности события
e5ac823f-c4f6-48e7-bebe-e6d44c57c7ad	Сводка
66aa3a39-c866-4f34-9e99-e75f9918eae7	Комментарии оператора
99914915-c882-4d11-80ff-57acdc6cc015	Заголовок
2f4a647e-4d9e-48ad-bf11-b1e49ffeac7f	Сообщение
1bf8a893-7d21-4c0c-9a2d-2e333a2d769d	Полное имя субъекта
0de358e0-c91b-4333-b902-000000000003	Фамилия / Номер авто
0de358e0-c91b-4333-b902-000000000001	Имя / Модель авто
0de358e0-c91b-4333-b902-000000000002	Отчество / Цвет авто
0de358e0-c91b-4333-b902-000000000006	Табельный номер субъекта
0de358e0-c91b-4333-b902-000000000009	Организация
0de358e0-c91b-4333-b902-000000000004	Подразделение
0a679144-d5ce-476d-a56e-0a696f079b71	Описание подразделения
0de358e0-c91b-4333-b902-00000000000a	Группа доступа
0de358e0-c91b-4333-b902-000000000005	Код карты
0de358e0-c91b-4333-b902-000000000007	Время действия идентификатора с
0de358e0-c91b-4333-b902-000000000008	Время действия идентификатора по
3ad06d24-43f6-45e0-8164-a98b4da955dc	Фото субъекта доступа
6fcfa1bb-9624-4248-a2d5-aa84901c53c8	Субъект в чёрном списке (флаг)
68d13785-c708-4418-8683-678a3f74957b	Изображения, прикрепленные к событию

- Если Вам необходимо подключить внешнюю библиотеку (dll), то в скрипте нужно написать следующий фрагмент кода в первых строках файла:

```
//css_ref <имя файла библиотеки>
```

Пример:

```
//css_ref System.Data.DataSetExtensions.dll  
//css_ref System.Web.dll
```

Сами файлы библиотек при этом нужно расположить в папке установки PNSoft, где расположен исполняемый файл сервиса “ParsecNET 3 Tasks”, по умолчанию это папка C:\Program Files\MDO\ParsecNET 3\

Либо же необходимо прописать папки, где расположены библиотеки, в переменной окружения PATH.