

Руководство по использованию комплекса программ xPipe



XPIPE: RUN BEFORE FLIGHT

Оглавление

<u>Введение</u>	2
<u>Системные требования</u>	3
<u>Установка</u>	4
<u>Установка xPipeServer</u>	4
<u>Установка xPipeViewer</u>	4
<u>Установка xPipeConnector</u>	4
<u>Установка xPipeDemo</u>	5
<u>Настройка и работа</u>	5
<u>Настройка и работа с xPipeServer</u>	6
<u>Настройка и работа с xPipeViewer</u>	8
<u>Настройка и работа с xPipeConnector</u>	17
<u>Особенности работы с xPipeViewer и MFSCClient/xPipeConnector</u>	25
<u>Подготовка к полетам с xPipeConnector</u>	25
<u>История изменений документа</u>	29
<u>Благодарности</u>	29
<u>Контакты</u>	30

Введение

Комплекс программ «xPipe» предназначен для получения от авиасимулятора X-Plane данных, представленных в «datarefs» (далее — датарефах), и их дальнейшего отображения и использования. В комплекс программ «xPipe» (далее — xPipe) входят:

- плагин для X-Plane, он же xPipeServer, выполняющий роль сервера данных;
- программа xPipeViewer, подключающаяся к xPipeServer и служащая для отображения состояния датарефов;
- набор библиотек xPipeConnector, позволяющих программе MFSCClient, являющейся клиентом ресурса www.virtairlines.ru / www.virtairlines.com, получать доступ к датарефам авиасимулятора X-Plane;
- исходные коды демонстрационной программы xPipeDemo на языке C#, выступающей в роли клиента xPipeServer и отображающей значения некоторых датарефов.

Сценарии возможного использования xPipe разнообразны. Можно просто использовать xPipeServer и xPipeViewer для того, чтобы узнать состояние интересующих Вас датарефов во время полета. Если Вы летаете в системе www.virtairlines.ru / www.virtairlines.com, Вы можете подключить её клиента MFSCClient к X-Plane через xPipeServer с помощью набор библиотек xPipeConnector, чтобы получить ряд преимуществ по сравнению с обычно используемым в X-Plane для этих целей XPUIPC. Также ничто не мешает Вам, имея опыт в программировании на C#, для получения датарефов из X-Plane написать собственного клиента для сервера xPipeServer.

Системные требования

- Операционная система Microsoft Windows (xPipe разрабатывался и тестировался на Microsoft Windows 10);
- NET Framework версии 3.5 или выше (скорее всего он уже установлен при включенных обновлениях операционной системы, если же по какой-то причине не установлен — его можно скачать по [этой ссылке](#));
- xPipe тестировался в работе с авиасимулятором X-Plane версии 11.26 (64-битная версия) , предполагается, что и в последующих версиях X-Plane xPipe будет успешно выполнять свои функции;
- набор библиотек xPipeConnector тестировался в работе с MFSCClient версии 2.2.8.1. Невозможно быть полностью уверенным в том, что xPipeConnector будет работать с последующими версиями MFSCClient, остается лишь надеяться на это.

Программы, входящие в xPipe, весьма нетребовательны к ресурсам компьютера: по результатам тестов среднее потребление времени процессора — около 1%.

Установка

xPipe поставляется в zip-архиве. Вам нужно распаковать его содержимое в любое удобное для Вас место. Архив состоит из каталогов:

- xPipeServer
- xPipeViewer
- xPipeConnector
- xPipeDemo

Далее Вам будет необходимо скопировать содержимое этих каталогов в зависимости от того, что из xPipe предполагается использовать, о чем будет написано ниже.

Установка xPipeServer

Каталог xPipeServer необходимо скопировать в место, где размещаются плагины X-Plane. Обычно это <каталог X-Plane>\Resources\plugins

Установка xPipeViewer

Программа xPipeViewer является «портативной» (portable), поэтому не имеет значения, где она расположена. Вы можете скопировать каталог xPipeViewer в любое удобное для Вас место. Для удобства использования рекомендуется создать на рабочем столе ярлык для xPipeViewer.exe.

Установка xPipeConnector

Перед установкой xPipeConnector настоятельно рекомендуется сделать

резервную копию каталога программы MFSCClient (обычно расположен в [«C:\Program Files\MFSCClient»](#)) или, по крайней мере, резервную копию файла FSUIPCClient.dll, содержащегося в каталоге программы. Этот файл отвечает за обмен информацией между программой MFSCClient и XPUIPC (это плагин для X-Plane, эмулирующий протокол работы FSUIPC для Microsoft Flight Simulator). Если по каким-то причинам Вам не подойдет xPipe, Вы всегда сможете вернуть из резервной копии файл FSUIPCClient.dll на его исходное место, и тем самым вернуться к стандартному способу работы программы MFSCClient.

После того, как все необходимые резервные копии сделаны, содержимое каталога xPipeConnector (именно содержимое, но не каталог целиком!) нужно скопировать в каталог программы MFSCClient (обычно расположен в [«C:\Program Files\MFSCClient»](#)). В процессе копирования уже существующий файл FSUIPCClient.dll будет заменен на новый файл (при копировании Вам потребуется подтвердить перезапись) из каталога xPipeConnector.

Установка xPipeDemo

xPipeDemo – каталог, содержащий в себе проект для Microsoft Visual Studio 2017. Если Вы собираетесь написать программу для взаимодействия с xPipeServer или хотите получить представление о том, как это всё работает, то этот проект будет Вам полезен. На наглядном примере Вы сможете изучить принципы организации обмена данными между xPipeServer и программой-клиентом. В противном случае — просто игнорируйте xPipeDemo.

Настройка и работа

Все программы, входящие в xPipe, хранят настройки для своей работы в файлах settings.xml, расположенных обычно в тех же каталогах, что и сами

программы. Настройки считываются единожды во время запуска программ, поэтому если Вы изменили какие-то значения настроек, до перезапуска программ они не вступят в силу. Перезапуск большинства программ xPipe производится очевидным способом, но xPipeServer является плагином для X-Plane, поэтому его можно перезапустить так:

- перезапустить X-Plane (долгий способ);
- с помощью штатных средств X-Plane остановить плагин xPipeServer, а потом снова его запустить (предпочтительный способ).

Настройка и работа с xPipeServer

Настройки xPipeServer хранятся в файле <каталог X-Plane>\Resources\plugins\xPipeServer\settings.xml. Ниже приведено содержание такого файла по-умолчанию:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <settings>
    <aquisitionfrequency value="10"/>
    <maxconnections value="2"/>
  </settings>
</root>
```

У xPipeServer есть два параметра настройки: **aquisitionfrequency** и **maxconnections**. Рассмотрим каждый из них подробнее:

- **aquisitionfrequency** — частота считывания датарифов в Гц (количество раз в секунду). Возможные значения: целые числа в диапазоне от 1 до 50. По-умолчанию значение равно 10 Гц. Тесты показали, что такое значение достаточно для большинства применений xPipe. Важно отметить, что это — частота чтения датарифов xPipeServer, но нет никаких гарантий того, что клиенты xPipeServer будут получать данные с точно такой же

частотой, поскольку непосредственно на передачу данных клиенту может влиять множество посторонних факторов. *Обычно* клиенты получают данные от xPipeServer на заявленной частоте считывания датарефов, но *иногда* эта частота может снижаться приблизительно на 10%.

- **maxconnections** – максимальное количество одновременно подключенных к xPipeServer клиентов. Возможные значения: целые числа в диапазоне от 1 до 64. По-умолчанию значение равно 2. Вы можете изменять значение в допустимом диапазоне, но не рекомендуется завышать его без явной необходимости, так как на каждое соединение (даже неиспользуемое) с клиентом xPipeServer расходует некоторое, пусть и пренебрежимо малое, количество вычислительных мощностей компьютера.

xPipeServer начинает работу непосредственно после запуска X-Plane, даже без запущенного полета. Пока полет не начат (самолет не размещен на месте отправления), значения датарефов, которые могут быть считаны xPipeServer, носят неопределенный характер. xPipeServer не берет на себя функцию осмысления содержания данных датарефов, эта задача целиком и полностью лежит на плечах клиентов. Из этого следует, что любых клиентов, работающих с xPipeServer, лучше всего запускать после того, как полет уже начат.

В момент запуска (а также — в момент остановки) xPipeServer использует штатный лог-файл X-Plane для вывода сервисных сообщений о своем состоянии. Если xPipeServer по каким-то причинам не смог запуститься, для анализа проблем в лог-файле X-Plane следует искать записи, начинающиеся со строки «xPipeServer: ».

Если запуск прошел успешно, далее всю информацию о своей работе xPipeServer пишет в файл xPipeServer.log, расположенный в каталоге плагина (<каталог X-Plane>\Resources\plugins\xPipeServer). Этот файл перезаписывается

при каждом запуске xPipeServer.

Для взаимодействия с клиентами xPipeServer использует технологию именованных каналов (named pipes), открывая для обмена данными канал «xpipe». Если по каким-то невероятным причинам на компьютере будет запущено одновременно более одного экземпляра X-Plane, работоспособным может быть только xPipeServer из экземпляра X-Plane, запущенного первым.

Настройка и работа с xPipeViewer

Настройки xPipeViewer хранятся в файле settings.xml, расположенном в каталоге программы. Ниже приведено типовое содержание такого файла:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <settings>
    <serveraddress value="."/>
    <writeflightlog value="1"/>
  </settings>
  <datarefs>
    <acficao path="sim/aircraft/view/acf_ICAO" type="s"/>
    <groundspeed path="sim/flightmodel/position/groundspeed" type="f"/>
    <onground path="sim/flightmodel/failures/onground_any" type="i"/>
    <latitude path="sim/flightmodel/position/latitude" type="d"/>
    <gear path="sim/flightmodel2/gear/deploy_ratio" type="f[]"/>
    <burn path="sim/flightmodel2/engines/engine_is_burning_fuel" type="i[]"/>
  </datarefs>
</root>
```

Настройки сгруппированы по двум разделам: [settings](#) и [datarefs](#). Количество параметров раздела [settings](#) неизменно, но раздел [datarefs](#) может включать в себя произвольное количество «записей», относящихся к датарефам, данные которых мы хотели бы видеть отображенными в программе xPipeViewer. По этой причине выше указано, что это — «типовое» содержание файла settings.xml: раздел [datarefs](#) может быть сколь угодно большим или пустым (что не имеет практического смысла, но в принципе такое возможно).

У xPipeViewer в разделе [settings](#) есть два параметра настройки: [serveraddress](#) и [writeflightlog](#). Рассмотрим каждый из них подробнее:

- [serveraddress](#) — адрес расположения xPipeServer. Если xPipeViewer и xPipeServer размещаются на одном и том же компьютере, значение этого параметра должно равняться «.» (точка). *Отметим, что технология именованных каналов (named pipes), используемая для связи между xPipeServer и клиентами (здесь клиент — xPipeViewer), допускает передачу данных по сети, но сетевой режим работы xPipe не тестировался, а задача настройки сетевого окружения для работоспособности такого режима передачи данных выходит за рамки этого руководства. Предполагается, что в случае если xPipeViewer и xPipeServer размещаются на разных компьютерах, значение параметра нужно установить равным сетевому имени компьютера (не IP-адресу!) с xPipeServer, но еще раз — такой режим работы xPipe не проверялся.*
- [writeflightlog](#) — признак ведения лога полета («чёрного ящика»). Если значение параметра равно 1, то после запуска программы в подкаталоге «FlightLogs» будет создан файл с именем, соответствующим дате и времени запуска программы, и расширением «.log». В этот файл xPipeViewer будет записывать все получаемые от xPipeServer данные датарефов вплоть до завершения своей работы. В первую строку файла записывается список псевдонимов датарефов из раздела настроек [datarefs](#) (об этом — будет написано далее), разделенных символом табуляции. Начиная со второй строки в файл будут записаны дата и время получения данных, а так же все фактически полученные данные, в порядке, соответствующем списку псевдонимов датарефов, разделенные символом табуляции. По окончании работы программы полученный log-файл можно открыть любым подходящим редактором таблиц (Microsoft Excel, OpenOffice Calc и т.д.), указав при этом символ табуляции как разделитель

колонок, для анализа полета. Итоговый размер log-файла зависит от количества датарефов, частоты получения данных и длительности работы программы. Будьте осторожны с включением этой настройки — размер log-файла может оказаться значительным!

Вернемся к разделу настроек [datarefs](#). Здесь перечисляются все датарефы, данные которых мы хотели бы видеть отображенными в программе xPipeViewer. На примере одного из них изучим описание формата датарефа, получение которого xPipeViewer запросит у xPipeServer:

```
... <acfiction path="sim/aircraft/view/acf_ICAO" type="s"/>
...
```

Описание датарефа состоит из трех частей (соответственно цветам):

- **псевдоним датарефа** – идентификатор датарефа внутри xPipeViewer, также, если **writeflightlog** = «1», используется для вывода в файл лога полета («чёрный ящик») в качестве названия колонки данных. Наименование может быть любым (с учётом ограничений xml-формата), но оно обязательно должно быть уникальным в рамках раздела настроек [datarefs](#), это очень важно!
- **path** — указывает на, собственно, адрес датарефа в X-Plane. Один и тот же адрес датарефа может быть использован в нескольких описаниях датарефов в xPipeViewer.
- **type** — один из нескольких поддерживаемых xPipe форматов датарефов. В большинстве случаев датареф X-Plane имеет однозначно известный тип, указанный в SDK X-Plane, но xPipeViewer не обладает этой информацией, поэтому Ваша задача — указать корректный тип данных.

xPipe поддерживает следующие типы данных:

- `i` – 32-битное знаковое целое число (`integer`);
- `f` – число с плавающей точкой одинарной точности (`float`);
- `d` – число с плавающей точкой двойной точности (`double`);
- `s` – строка символов (`string`);
- `i[]` – одномерный массив произвольной длины, состоящий из 32-битных знаковых целых чисел (`integer`);
- `f[]` – одномерный массив произвольной длины, состоящий из чисел с плавающей точкой одинарной точности (`float`).

В SDK X-Plane существуют датарефы, представленные многомерными массивами, но работу с такими типами данных xPipe не поддерживает.

Вернемся к разделу настроек `datarefs` исходного файла `settings.xml` и выясним, что же здесь xPipeViewer будет запрашивать у xPipeServer:

```
<datarefs>
  <acficao path="sim/aircraft/view/acf_ICAO" type="s"/>
  <groundspeed path="sim/flightmodel/position/groundspeed" type="f"/>
  <onground path="sim/flightmodel/failures/onground_any" type="i"/>
  <latitude path="sim/flightmodel/position/latitude" type="d"/>
  <gear path="sim/flightmodel2/gear/deploy_ratio" type="f[]"/>
  <burn path="sim/flightmodel2/engines/engine_is_burning_fuel" type="i[]"/>
</datarefs>
```

1. Псевдоним `acficao`, датареф `"sim/aircraft/view/acf_ICAO"`, тип данных — строка символов. Это код ICAO модели самолета, загруженного в X-Plane.
2. Псевдоним `groundspeed`, датареф `"sim/flightmodel/position/groundspeed"`, тип данных — число с плавающей точкой одинарной точности. Это текущая скорость самолета относительно земли.
3. Псевдоним `onground`, датареф `"sim/flightmodel/failures/onground_any"`, тип

данных — 32-битное знаковое целое число. Это положение самолета относительно земли: 0 — самолет в воздухе, 1 — самолет хотя бы одной из стоек касается земли.

4. Псевдоним `latitude`, датареф `"sim/flightmodel/position/latitude"`, тип данных — число с плавающей точкой двойной точности. Это одна из текущих географических координат самолета относительно земли — широта.
5. Псевдоним `gear`, датареф `"sim/flightmodel2/gear/deploy_ratio"`, тип данных — одномерный массив чисел с плавающей точкой одинарной точности. Это текущее положение каждой из стоек шасси самолета: 0 — полностью убрано, 1 — полностью выпущено.
6. Псевдоним `burn`, датареф `"sim/flightmodel2/engines/engine_is_burning_fuel"`, тип данных — одномерный массив 32-битных знаковых целых чисел. Это текущее состояние двигателей самолета: 0 — двигатель не сжигает топливо, 1 — двигатель сжигает топливо.

Итак, мы выяснили, что означают настройки, записанные в `settings.xml` `xPipeViewer`. Пришло время для знакомства с интерфейсом программы. Запустим `X-Plane`, но пока не будем начинать полет, оставшись на стартовом экране, и после этого запустим `xPipeViewer` (если `X-Plane` запущен от имени администратора, то `xPipeViewer` тоже должен запускаться от имени администратора!). При запуске `xPipeViewer` самостоятельно пытается установить связь с `xPipeServer`, Вам не нужно предпринимать для этого никаких действий. В случае обрыва связи `xPipeViewer` также будет самостоятельно пытаться восстановить связь. После запуска и установления связи мы увидим следующую картинку:

The screenshot shows the xPipeViewer 0.1 application window. It features a menu bar with 'File' and 'About'. The main area contains a table with the following data:

Name	DataRef path	DataRef type	Value
acfticao	sim/aircraft/view/acf_ICAO	String	
groundspeed	sim/flightmodel/position/groundspeed	Float	0.0000
onground	sim/flightmodel/failures/onground_any	Integer	0
latitude	sim/flightmodel/position/latitude	Double	0.00000000
gear	sim/flightmodel2/gear/deploy_ratio	Float[...]	0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000
bum	sim/flightmodel2/engines/engine_is_burning_fuel	Integer[...]	0,0,0,0,0,0,0

Below the table is a log window showing the following messages:

```

13:27:34 Communication established
13:27:34 Connecting...
13:27:34 Connecting xPipeServer...
13:27:34 Reading program settings
13:27:34 xPipeViewer 0.1 started

```

At the bottom of the window, a status bar displays 'Communication established' and '10 Hz'.

Интерфейс xPipeViewer стандартен (сверху — вниз):

- системное меню;
- таблица с данными (колонки: псевдоним, датареф, тип данных, текущее значение);
- окно с логом нескольких последних событий;
- статусная полоса, указывающая состояние соединения с xPipeServer и (если связь установлена) частоту приема данных.

Если связь с xPipeServer еще не установлена, колонка текущих значений для всех псевдонимов будет содержать значение «NO DATA» (после обрыва связи в колонке текущих значений будут оставаться данные, полученные последними). В нашем примере связь установлена, поэтому мы видим в текущих значениях либо «0», либо пустое значение — полет еще не начат, и xPipeServer транслирует «неопределенные» значения.

Запустим полет в X-Plane, выбрав для этого Сессну 172 и произвольный аэропорт. После загрузки полета в колонке текущих значений появились данные:

The screenshot shows a window titled 'xPipeViewer 0.1' with a menu bar containing 'File' and 'About'. Below the menu bar is a table with the following data:

Name	DataRef path	DataRef type	Value
acficao	sim/aircraft/view/acf_ICAO	String	C172
groundspeed	sim/flightmodel/position/groundspeed	Float	0.0001
onground	sim/flightmodel/failures/onground_any	Integer	1
latitude	sim/flightmodel/position/latitude	Double	59.72320956
gear	sim/flightmodel2/gear/deploy_ratio	Float[...]	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000
burn	sim/flightmodel2/engines/engine_is_burning_fuel	Integer[...]	0,0,0,0,0,0,0

Below the table is a log of messages:

```

13:45:28 X-Plane message received: XPLM_MSG_AIRPORT_LOADED (103)
13:45:19 X-Plane message received: XPLM_MSG_SCENERY_LOADED (104)
13:44:35 Communication established
13:44:35 Connecting...
13:44:35 Connecting xPipeServer...
13:44:35 Reading program settings
  
```

At the bottom of the window, it says 'Communication established' and '9 Hz'.

Немного изменим размеры формы xPipeViewer так, чтобы не отображалось пустое место в таблице данных. Запустим двигатель нашей Сессны. Обратите внимание, что данные псевдонима «burn» изменились — было «0, 0, 0, 0, 0, 0, 0, 0», стало «1, 0, 0, 0, 0, 0, 0, 0». На Сессне установлен один двигатель, и датареф отобразил изменение его состояния:

Name	DataRef path	DataRef type	Value
acficao	sim/aircraft/view/acf_ICAO	String	C172
groundspeed	sim/flightmodel/position/groundspeed	Float	0.0001
onground	sim/flightmodel/failures/onground_any	Integer	1
latitude	sim/flightmodel/position/latitude	Double	59.72320963
gear	sim/flightmodel2/gear/deploy_ratio	Float[...]	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000
bum	sim/flightmodel2/engines/engine_is_burning_fuel	Integer[...]	1,0,0,0,0,0,0

13:46:58 X-Plane message received: XPLM_MSG_WILL_WRITE_PREFS (107)
13:45:28 X-Plane message received: XPLM_MSG_AIRPORT_LOADED (103)

Communication established 10 Hz

Теперь совершим взлет. Изменились данные псевдонимов «groundspeed», «onground», «latitude», что неудивительно — мы набрали скорость, оторвались от земли, и, пролетев некоторое расстояние, изменили географическую координату широта нашего самолета. Данные же псевдонима «gear» остались без изменений — у Сесны 172 шасси не убираются.

Name	DataRef path	DataRef type	Value
acficao	sim/aircraft/view/acf_ICAO	String	C172
groundspeed	sim/flightmodel/position/groundspeed	Float	38.9034
onground	sim/flightmodel/failures/onground_any	Integer	0
latitude	sim/flightmodel/position/latitude	Double	59.73498569
gear	sim/flightmodel2/gear/deploy_ratio	Float[...]	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000
bum	sim/flightmodel2/engines/engine_is_burning_fuel	Integer[...]	1,0,0,0,0,0,0

13:46:58 X-Plane message received: XPLM_MSG_WILL_WRITE_PREFS (107)
13:45:28 X-Plane message received: XPLM_MSG_AIRPORT_LOADED (103)

Communication established 9 Hz

Завершим вылет, закрыв X-Plane, и закроем xPipeViewer. Поскольку в настройках параметр **writeflightlog** был установлен в значение «1», в подкаталоге FlightLogs был создан файл лога полета («чёрный ящик») с именем 181101134435.log. Имя этого файла дает нам понять, когда он начал наполняться данными (слева — направо): 18-й год, 11-й месяц (ноябрь), 01 число, время — 13:44:35 (это дата и время написания этой главы руководства). Вы можете

открыть этот файл в любом табличном редакторе как текстовый файл с разделителем колонок «табуляция» для анализа параметров полета:

	A	B	C	D	E	F	G
1	18.11.01 13:44:35	acficao	groundspeed	onground	latitude	gear	burn
3951	18.11.01 13:51:51	C172	33.8528		159.72593044	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3952	18.11.01 13:51:51	C172	33.9848		159.72595790	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3953	18.11.01 13:51:51	C172	34.1203		159.72598552	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3954	18.11.01 13:51:52	C172	34.2565		159.72601318	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3955	18.11.01 13:51:52	C172	34.4321		159.72605029	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3956	18.11.01 13:51:52	C172	34.5634		159.72607821	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3957	18.11.01 13:51:52	C172	34.7393		159.72611558	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3958	18.11.01 13:51:52	C172	34.8679		159.72614373	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3959	18.11.01 13:51:52	C172	34.9964		159.72617196	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3960	18.11.01 13:51:52	C172	35.1647		159.72620971	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3961	18.11.01 13:51:52	C172	35.2494		159.72622865	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3962	18.11.01 13:51:52	C172	35.4139		159.72626668	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3963	18.11.01 13:51:53	C172	35.5350		159.72629530	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3964	18.11.01 13:51:53	C172	35.6887		159.72633360	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3965	18.11.01 13:51:53	C172	35.7974		159.72636243	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3966	18.11.01 13:51:53	C172	35.8983		059.72639133	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3967	18.11.01 13:51:53	C172	36.0211		059.72642998	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3968	18.11.01 13:51:53	C172	36.0922		059.72645420	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3969	18.11.01 13:51:53	C172	36.1854		059.72648817	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3970	18.11.01 13:51:53	C172	36.2605		059.72651736	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3971	18.11.01 13:51:53	C172	36.3318		059.72654662	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0
3972	18.11.01 13:51:54	C172	36.4211		059.72658571	1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000,1.0000	1,0,0,0,0,0,0,0

Здесь хорошо видно, когда значение псевдонима «onground» сменилось с «0» на «1» — в этот момент был произведен взлет.

Внимательный читатель этого руководства мог заметить, что в правом нижнем углу формы xPipeViewer частота получения данных изменялась: в некоторых случаях 9 Гц, в некоторых — 10 Гц. Периодическое кратковременное снижение частоты приема данных на величину около 10% считается нормальным (см. раздел руководства «[Настройка и работа xPipeServer](#)»).

Помимо датарефов xPipeViewer принимает от xPipeServer системные сообщения X-Plane, касающиеся состояния самого авиасимулятора: загрузка модели самолета, ливреи, региона, сценария аэропорта, возникновение катастрофы и прочие. Принятие такого рода сообщений регистрируется в окне лога последних событий и в файле лога программы — xPipeViewer.log.

Таким образом зная нужные вам датарефы Вы всегда можете получить из них данные с помощью xPipeViewer. Список и краткое описание всех поддерживаемых в текущей версии X-Plane датарефов Вы можете найти в <каталог X-Plane>\Resources\plugins\DataRefs.txt

Настройка и работа с xPipeConnector

Перед чтением этой главы руководства убедитесь, что Вы внимательно прочитали предыдущую главу «[Настройка и работа с xPipeViewer](#)». Дело в том, что xPipeConnector хранит свои настройки в том же формате, что и xPipeViewer, и в целом он не слишком сильно отличается от xPipeViewer по своему внутреннему устройству. Разница между ними заключается в следующем:

- Настройки xPipeConnector по сравнению с xPipeViewer имеют дополнительные разделы.
- список и описания датарефов в разделе настроек [datarefs](#) не рекомендуется изменять (особенно — удалять элементы списка!), так как это может повлечь неработоспособность программы. Но в случае необходимости Вы можете добавлять этот список новые элементы.
- xPipeConnector не имеет своей формы для отображения данных датарефов, так как передает их в программу MFSCClient.

Из перечисленного выше следует, что Вы можете использовать файл настроек xPipeConnector при работе с xPipeViewer, если это будет нужно — xPipeViewer при считывании настроек проигнорирует незнакомые ему разделы. Но обратное неверно — не следует использовать файл настроек xPipeViewer в xPipeConnector, так как такой файл не будет содержать необходимых для работы xPipeConnector разделов!

Настройки xPipeConnector хранятся в файле settings.xml, расположенном в каталоге программы MFSCClient (обычно расположен в «[C:\Program Files\MFSCClient](#)»).

Рассмотрим типовое содержание этого файла:

```

<?xml version="1.0" encoding="utf-8" ?>
<root>
  <settings>
    <serveraddress value="."/>
    <writeflightlog value="1"/>
  </settings>
  <flight>
    <stopflightonenginesoff value="1"/>
    <freezevpmontouchdown value="2000"/>
  </flight>
  <aliases>
    <alias0 acficao="AN24" virtairlines="An-24RV"/>
    <alias1 acficao="YK40" virtairlines="Yak-40"/>
  </aliases>
  <redefinitions>
    <redefinition0 acficao="AN24" file="redefinitions\An24.xml"/>
    <redefinition1 acficao="YK40" file="redefinitions\Yak-40.xml"/>
  </redefinitions>
  <datarefs>
    <acficao path="sim/aircraft/view/acf_ICAO" type="s"/>
    <airspeed path="sim/cockpit2/gauges/indicators/airspeed_kts_pilot"
type="f"/>
    <groundspeed path="sim/flightmodel/position/groundspeed" type="f"/>
    <localsec path="sim/time/local_time_sec" type="f"/>
    <zulusec path="sim/time/zulu_time_sec" type="f"/>
    <day path="sim/time/local_date_days" type="i"/>
    <msl path="sim/flightmodel/position/elevation" type="d"/>
    <latitude path="sim/flightmodel/position/latitude" type="d"/>
    <longitude path="sim/flightmodel/position/longitude" type="d"/>
    <vfpmpath="sim/flightmodel/position/vh_ind_fpm" type="f"/>
    <onground path="sim/flightmodel/failures/onground_any" type="i"/>
    <simrate path="sim/time/sim_speed" type="i"/>
    <lighttaxi path="sim/cockpit/electrical/taxi_light_on" type="i"/>
    <lightland path="sim/cockpit/electrical/landing_lights_on" type="i"/>
    <lightbeacon path="sim/cockpit/electrical/beacon_lights_on" type="i"/>
    <lightnav path="sim/cockpit/electrical/nav_lights_on" type="i"/>
    <lightstrobe path="sim/cockpit/electrical/strobe_lights_on" type="i"/>
    <gear path="sim/flightmodel2/gear/deploy_ratio" type="f[]"/>
    <brake path="sim/cockpit2/controls/parking_brake_ratio" type="f" />
    <flaps path="sim/flightmodel/controls/flaprat" type="f"/>
    <windspeed path="sim/weather/wind_speed_kt" type="f"/>
    <winddirection path="sim/weather/wind_direction_deg" type="f"/>
    <pressure path="sim/weather/barometer_sealevel_inhg" type="f"/>
    <agl path="sim/flightmodel/position/y_agl" type="f"/>
    <fuelkgmax path="sim/aircraft/weight/acf_m_fuel_tot" type="f"/>
    <fuelkg path="sim/flightmodel/weight/m_fuel_total" type="f"/>
    <fuelburn path="sim/flightmodel2/engines/engine_is_burning_fuel"
type="i[]"/>
  </datarefs>
</root>

```

Разделы настроек **settings** и **datarefs** полностью аналогичны таковым для программы xPipeViewer (за исключением параметра **writeflightlog** — признака ведения лога полета («чёрного ящика»). Если значение параметра равно 1, то после запуска программы в подкаталоге «FlightLogs» будут созданы два файла с

именем, соответствующим дате и времени запуска программы, и расширением «.log». Файл с окончанием «-in» содержит данные, полученные от сервера, а файл с окончанием «-out» содержит данные, отправленные в программу MFSCClient.). Добавлены три новых раздела: [flight](#), [aliases](#), [redefinitions](#) — далее мы рассмотрим их подробнее.

Раздел настроек [flight](#) и входящие в него параметры корректируют восприятие программой MFSCClient лётной ситуации.

```
...  
<flight>  
  <stopflightonenginesoff value="1"/>  
  <freezevfpmontouchdown value="2000"/>  
</flight>  
...
```

Рассмотрим каждый из параметров раздела:

- [stopflightonenginesoff](#) — влияет на условие завершения полёта. Если значение равно «1», то помимо установки самолета на стояночный тормоз для завершения полёта необходимо, чтобы все двигатели были выключены. Вы можете установить значение этого параметра по своему выбору («0» или «1»).
- [freezevfpmontouchdown](#) — устанавливает длительность «заморозки» величины вертикальной скорости самолёта в момент касания земли, значение параметра задается в миллисекундах. Дело в том, что программа MFSCClient считывает данные от авиасимулятора приблизительно раз в секунду, но xPipeServer обычно передает клиентам данные гораздо чаще (при настройках по-умолчанию — 10 раз в секунду), поэтому возможна ситуация считывания устаревших данных. В результате вертикальная скорость с точки зрения программы MFSCClient может оказаться отличной от той, что была на самом деле в момент касания земли. «Заморозка» величины вертикальной скорости самолёта в момент касания земли на 2 секунды (2000 миллисекунд) гарантирует, что программа MFSCClient

получит корректные данные.

Раздел настроек **aliases** предназначен для списка «соответствий» кодов ICAO моделей самолетов X-Plane и типов самолетов, используемых в работе программой MFSCClient. Для многих моделей самолетов X-Plane эти данные не совпадают, тогда для правильного распознавания модели самолета программой MFSCClient достаточно внести в раздел **aliases** элемент «соответствия».

```
...
<aliases>
  <alias0 acficao="AN24" virtairlines="An-24RV"/>
  <alias1 acficao="YK40" virtairlines="Yak-40"/>
</aliases>
...
```

Описание элемента «соответствия» состоит из трех частей (соответственно цветам):

- **aliasxxx** – идентификатор «соответствия» внутри xPipeConnector. Наименование может быть любым (с учётом ограничений xml-формата), но оно обязательно должно быть уникальным в рамках раздела настроек **aliases**, это очень важно!
- **acficao** — код ICAO модели самолета в X-Plane. Этот код можно узнать, открыв в текстовом редакторе acf-файл модели самолета и произведя поиск текста «acf/_ICAO» (без кавычек). Следующим текстовым фрагментом после «acf/_ICAO» и будет искомый код. Так же этот код можно узнать с помощью xPipeViewer, внося в список датарефов датареф "sim/aircraft/view/acf_ICAO" и загрузив полет с интересующим Вас самолетом.
- **virtairlines** — тип самолета, используемый программой MFSCClient. Тип интересующего Вас самолета можно выяснить на сайте www.virtairlines.ru (или www.virtairlines.com). Для, например, Beechcraft 1900D, типом

самолета будет значение «1900D»

Beechcraft 1900D	
Производитель:	Beechcraft
Тип:	1900D
Пассажиры:	19

Раздел настроек [redefinitions](#) предназначен для списка «переопределений» описаний датарефов xPipeConnector. Смысл таких «переопределений» заключается в следующем: возможна ситуация, когда для определенных моделей самолетов X-Plane датарефы, описанные в разделе настроек [datarefs](#), не будут содержать нужную информацию. В этом случае Вы можете для таких моделей самолетов переопределить источники датарефов на те, что будут содержать нужную информацию. При этом для остальных моделей самолетов источники датарефов останутся прежними.

```
...  
<redefinitions>  
  <redefinition0 acficao="AN24" file="redefinitions\An24.xml"/>  
  <redefinition1 acficao="YK40" file="redefinitions\Yak-40.xml"/>  
</redefinitions>  
...
```

Описание элемента «переопределения» состоит из трех частей (соответственно цветам):

- **redefinitionxxx** – идентификатор «переопределения» внутри xPipeConnector. Наименование может быть любым (с учётом ограничений xml-формата), но оно обязательно должно быть уникальным в рамках раздела настроек [redefinitions](#), это очень важно!
- **acficao** — код ICAO модели самолета в X-Plane, нуждающегося в переопределении источников датарефов. Этот код можно узнать, открыв в текстовом редакторе acf-файл модели самолета и произведя поиск текста «acf/_ICAO» (без кавычек). Следующим текстовым фрагментом после

«acf/_ICAO» и будет искомым код. Так же этот код можно узнать с помощью xPipeViewer, внося в список датарефов датареф "sim/aircraft/view/acf_ICAO" и загрузив полет с интересующим Вас самолетом.

- **file** — относительный путь к файлу, содержащему переопределение источников датарефов. Обычно файлы с переопределениями располагаются в подкаталоге «redefinitions».

Файл с переопределением источников датарефов имеет структуру, идентичную структуре файла настроек settings.xml, но содержит только один раздел настроек — **datarefs**. Все правила наполнения раздела настроек **datarefs** полностью соответствуют тем, что действуют для этого раздела в файлах настроек для xPipeViewer и xPipeConnector (см. [«Настройка и работа с xPipeViewer»](#)).

Приведем пример, как это работает. Предположим, что по каким-то причинам Вам захотелось, чтобы можно было «включить» все системы освещения в Сессне 172 с помощью лишь одного переключателя «taxi lights». При этом Вы должны понимать, что они будут «включены» только с точки зрения программы MFSCClient, так как ей будут переданы такие данные от xPipeConnector. Для этого создайте файл C172.xml (имя файла может быть любым, «C172» взято для наглядности), расположите его в подкаталоге «redefinitions». Запишите в файл следующий текст:

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
  <datarefs>
    <lightland path="sim/cockpit/electrical/taxi_light_on" type="i"/>
    <lightbeacon path="sim/cockpit/electrical/taxi_light_on" type="i"/>
    <lightnav path="sim/cockpit/electrical/taxi_light_on" type="i"/>
    <lightstrobe path="sim/cockpit/electrical/taxi_light_on" type="i"/>
  </datarefs>
</root>
```

По содержанию текста видно, что для всех псевдонимов датарефов теперь задан один и тот же адрес датарефа — «sim/cockpit/electrical/taxi_light_on». Данные этого датарефа будут транслироваться в псевдонимы датарефов `lightland`, `lightbeacon`, `lightnav`, `lightstrobe`.

После этого Вам нужно модифицировать раздел настроек `redefinitions` файла настроек `settings.xml` `xPipeConnector`, добавив в него «переопределение» для Сессны 172. Назовите «переопределение», например, `redefinition2`, для `acficao` задайте значение «C172» (это код ICAO этой модели самолета в X-Plane), а в `file` задайте путь к файлу переопределений — «`redefinitions\C172.xml`». Теперь раздел настроек `redefinitions` будет выглядеть так:

```
...
<redefinitions>
  <redefinition0 acficao="AN24" file="redefinitions\An24.xml"/>
  <redefinition1 acficao="YK40" file="redefinitions\Yak-40.xml"/>
  <redefinition2 acficao="C172" file="redefinitions\C172.xml"/>
</redefinitions>
...
```

Теперь Вы можете запустить полет в X-Plane и в программе MFSCClient (предварительно совершив все необходимые процедуры по бронированию рейса и т.д. на сайте www.virtairlines.ru / www.virtairlines.com). Включите в самолете «taxi lights» и Вы увидите, что программа MFSCClient показывает включенными все системы освещения: посадочные и навигационные огни, маяк, стробы.

В этом примере показано простейшее применение «переопределений». Но для некоторых моделей самолетов со сложной внутренней логикой работы такого перенаправления данных от одних датарефов к другим может оказаться недостаточно. В этом случае без программирования в том или ином виде скорее всего не обойтись: например можно с помощью скрипта Lua или плагина извлекать нужные данные из модели самолета и передавать их в самостоятельно созданный датареф, ссылку на который уже использовать в `xPipeConnector`. Но

подробное описание таких приемов работы выходит за рамки этого руководства.

Мы уделили много места описанию настроек xPipeConnector, но описание работы с ним будет довольно кратким — xPipeConnector не имеет своего пользовательского интерфейса, он работает «прозрачно» для пользователя, программа MFSCClient просто загружает его и использует для приема данных от X-Plane. xPipeConnector получает от программы MFSCClient список интересующих ее данных, отправляет запрос на их регулярное получение (с учётом «переопределений», если таковые описаны в настройках) в xPipeServer и ожидает ответа. Получив ответ с данными от xPipeServer, xPipeConnector передает их в программу MFSCClient, предварительно преобразовав (с учетом своих настроек) их в понятный для программы MFSCClient формат.

Информацию о своем состоянии xPipeConnector сохраняет лог-файл xPipeConnector.log, находящийся в каталоге программы, этот файл перезаписывается при каждом запуске программы. Если в настройках включена функция «чёрного ящика», то все данные, полученные от xPipeServer в течение сеанса работы, записываются в файл лога полёта (см. [«Настройка и работа с xPipeViewer»](#)) в подкаталог «FlightLogs».

Во введении (см. [«Введение»](#)) было упомянуто, что xPipeConnector дает ряд преимуществ по сравнению с обычно используемым для связи между X-Plane и программой MFSCClient пакетом XPUIPC. Перечислим их:

- Известен и открыт набор датарефов, используемых для передачи в программу MFSCClient, и этот набор можно изменить (redefinitions, Lua и т.д.), если возникнет необходимость.
- Поддержка «соответствий» (aliases) позволяет летать в X-Plane с

программой MFSCClient без переименования каталогов моделей самолетов и/или прочего неочевидного «колдовства».

- «Заморозка» величины вертикальной скорости самолёта в момент касания земли гарантирует, что программа MFSCClient получит корректные данные.
- В программе MFSCClient отображается положение закрылок, корректно отображается состояние световой техники (для моделей самолетов, использующих соответствующие стандартные датарефы).
- Встроенная функция «чёрного ящика», позволяющая узнать все интересующие параметры полёта по его завершении.

Особенности работы с xPipeViewer и MFSCClient/xPipeConnector

Если для получения данных используется искусственно созданный датареф (например — с помощью скрипта Lua, стороннего программного обеспечения), важно, чтобы xPipeViewer и программа MFSCClient (работающая через xPipeConnector) были запущены строго после того, как все скрипты и т. д. отработали и уже создали искусственный датареф! Дело в том, что xPipeViewer и xPipeConnector в подавляющем большинстве сценариев своей работы посылают к серверу данных xPipeServer запрос со списком нужных датарефов лишь однажды, во время своего запуска. Поэтому, если при их запуске окажется, что датареф еще не существует, то во время всего сеанса работы xPipeViewer и xPipeConnector не смогут получать данные из этого датарефа (скорее всего данные будут выглядеть как «ERROR»).

Подготовка к полетам с xPipeConnector

Содержание данного раздела носит характер рекомендаций — как по

мнению автора практичнее всего приступить к полетам в системе www.virtairlines.ru / www.virtairlines.com, используя xPipe.

Поскольку для X-Plane создано и продолжает создаваться огромное количество различных моделей самолетов, невозможно гарантировать, что каждая из них сможет без дополнительных настроек, «из коробки», успешно работать с программой MFSCClient (а следовательно — с www.virtairlines.ru / www.virtairlines.com) через xPipe. В этих условиях разумным звучит предположение, что перед выполнением регулярных полётов необходимо удостовериться, что все необходимые данные о состоянии модели самолета передаются и корректно воспринимаются программой MFSCClient.

Предлагается следующий алгоритм действий для проверки модели самолета:

1. Создать и забронировать тренировочный полет с участием интересующего Вас самолета.
2. Завести список неисправностей (например — текстовый файл). Мы будем надеяться, что он останется пуст, но нужно быть готовым ко всему.
3. Загрузить X-Plane с самолетом в аэропорту вылета и программу MFSCClient, удостовериться в установлении связи между авиасимулятором и программой MFSCClient. Если связь не установилась, проверить лог-файлы xPipeServer и xPipeConnector, по результатам проверки устранить неисправность (возможно потребуется перезапустить программу MFSCClient и/или xPipeServer, если их настройки связи изменялись).
4. Удостовериться, что программа MFSCClient корректно распознала модель самолета. Если этого не произошло, добавить необходимое «соответствие» в раздел настроек [aliases](#) xPipeConnector и перезапустить программу MFSCClient. Повторить этот пункт действий до достижения положительного результата.

5. Запустить все системы (электрические, топливные, двигатели и все прочие), привести самолет в состояние готовности к вылету.
6. Проверить световую технику: в программе MFSCClient должно корректно отображаться включение и выключение всех нужных световых приборов. Если состояние некоторых световых приборов не отображается в программе MFSCClient, отметить их в списке неисправностей.
7. Сравнить вес топлива на борту по мнению авиасимулятора и по мнению программы MFSCClient. Если разница больше 1...2%, отметить это в списке неисправностей.
8. Проверить выпуск и уборку закрылок — программа MFSCClient должна отображать изменение угла выпуска закрылок. Угол выпуска по мнению авиасимулятора и по мнению программы MFSCClient может отличаться, это не играет важной роли. Если программа MFSCClient не отображает изменение угла выпуска закрылок, отметить это в списке неисправностей.
9. Проверить передачу данных о состоянии тормозов. Для этого необходимо воспользоваться xPipeViewer, при этом раздел настроек [datarefs](#) xPipeViewer должен в точности соответствовать разделу настроек [datarefs](#) xPipeConnector. Включите парковочный тормоз (давление 100%) и проверьте данные в псевдониме датарефа «brake» — значение должно равняться «1». Если условие не выполняется, отметить это в списке неисправностей.
10. Программа MFSCClient должна отображать текущее состояние шасси: на земле они должны быть выпущены. Если самолет имеет убирающиеся шасси — взлетайте и выполните их уборку. В случае, если после уборки программа MFSCClient не показывает, что шасси убраны — отметить это в списке неисправностей. Если взлет производился — совершите посадку.
11. В случае, если Вы установили настройку xPipeConnector

`stopflightonenginesoff` равной «1», то для завершения полета необходимо (помимо включенного парковочного тормоза) заглушить двигатели. Для проверки состояния двигателей воспользуйтесь xPipeViewer. При заглушенных двигателях псевдоним датарефа «fuelburn» (это массив чисел) должен содержать только значения «0». Если условие не выполняется, отметить это в списке неисправностей.

Если в списке неисправностей не оказалось ни одной записи — поздравляем, самолет готов к полетам в системе www.virtairlines.ru / www.virtairlines.com с использованием xPipe! Создавайте регулярные рейсы и летайте на этом самолете!

Что делать, если в списке неисправностей все же кое-что есть? Зависит от типа неисправности. К фатальным неисправностям, при которых Вы не сможете совершить полет в системе www.virtairlines.ru / www.virtairlines.com с использованием xPipe, относятся проблемы с парковочным тормозом и состоянием двигателей (только если Вы установили настройку `xPipeConnector stopflightonenginesoff` равной «1») — после приземления в порту назначения Вам не удастся завершить вылет. Неисправности со световой техникой, шасси, закрылками скорее всего приведут к штрафу за их неправильное использование. Неизвестно, к чему может привести неисправность, связанная со слишком большой разницей в весе топлива, но ничего хорошего это не сулит.

Как починить эти неисправности? К сожалению, здесь не может быть однозначного ответа: всё зависит от модели самолета. Если разработчики модели самолета не предусмотрели никакой возможности получения данных о внутреннем состоянии его систем, то скорее всего полеты на нём в системе www.virtairlines.ru / www.virtairlines.com с использованием xPipe будут либо невозможны, либо будут приводить к постоянным штрафам. Если же получить эти данные можно, то с помощью redefinitions, скриптов Lua и т.д. Вы сможете

вывести нужные для полета данные в xPipe — нужно лишь выяснить их расположение, способ организации (в этом может помочь xPipeViewer, документация по модели самолета, если таковая существует), но от Вас потребуется много труда для достижения успеха. Что именно нужно делать? К сожалению обсуждение этого вопроса из-за его обширности выходит за рамки этого руководства.

История изменений документа

- 11 июня 2019 года — изменена глава «[Настройка и работа с xPipeConnector](#)», внесено дополнение по параметру **writeflightlog**. Версия документа — 0.3.
- 21 ноября 2018 года — добавлена глава «[Особенности работы с xPipeViewer и MFSCClient/xPipeConnector](#)». Версия документа — 0.2.
- 5 ноября 2018 года — написание первой версии руководства завершено. Версия документа — 0.1.

Благодарности

От лица автора xPipe хочу сказать большое спасибо:

- Антону Шлаину — за основную идею, обсуждение концепции проекта, первичное тестирование бета-версии, а также разработку логотипа проекта.
- Михаилу Дубовику — за информационную поддержку, выделение места для проекта на сайте сообщества virtavia.online, помощь с дизайном и

разработкой сайта проекта.

- Сообществу виртуальных пилотов [VirtAvia](#) – за вдохновение, помощь и общение.

Контакты

Сайт проекта xPipe — xpipe.virtavia.online

Разработчик xPipe – Станислав Чанков, Вы можете связаться со мной по вопросам, касающимся xPipe, через [форум](#), [VK](#), [FaceBook](#).