



## Руководство администратора

## Введение

LWF — это клиентское приложение системы ЛОЦМАН:PLM для удобной работы с документами и бизнес-процессами.

Для работы LWF на предприятии должна быть установлена и правильно настроена система ЛОЦМАН:PLM. Документация по настройке комплекса находится в папке установки ЛОЦМАН:PLM, обычно это C:\Program Files (x86)\ASCON\Loodsman\Doc.

На данный момент LWF реализует не все функции ЛОЦМАН Клиент, в будущем количество реализованных функций будет увеличиваться.

## Состав приложения

Для работы LWF на компьютер пользователя должны быть установлены следующие файлы:

Имя файла	Обязательный	Описание
LUHost.exe	Нет	Изолированный запуск модулей просмотра и подключаемых модулей.
LUStub.dll	Да	Библиотека совместимости со старыми версиями ЛОЦМАН Клиент и старыми плагинами.
LWF.exe	Да	Основной исполняемый файл приложения.
LWF.menu	Нет	Файл с настройками меню для всех пользователей приложения. Путь и имя этого файла можно задать в файле конфигурации.
LWF.xml	Нет	Файл конфигурации. Путь и имя этого файла можно задать в командной строке, например: LWF.exe /C Organization.xml
midas.dll	Да	Библиотека для работы с пакетами данных, возвращаемыми сервером приложений.

Файлы обычно устанавливаются в папку C:\Program Files (x86)\ASCON\LWF, но возможна установка в любое другое место.

## Подключение к серверу

По умолчанию, для подключения к серверу приложений LWF использует те же настройки, что и ЛОЦМАН Клиент. Вы можете указать для LWF отдельные параметры подключения к серверу и базам данных в конфигурационном файле.

Параметры подключения к серверу и базам данных ЛОЦМАН Клиент хранятся в реестре, в ветке HKCU\Software\ASCON\Loodsman. В значениях SP, ClientSP или WorkFlowSP хранится список серверов приложений. Информация о параметрах подключения к базам данных хранится в значении DBList. Вы можете настроить эти параметры, воспользовавшись стандартным клиентом, выбрав в меню «Инструменты → Выбрать сервер приложений» и нажав кнопку «Настройка подключения к базам данных» на панели инструментов, после чего распространить эти параметры на все компьютеры организации.

## Файл конфигурации

Большинство настроек LWF хранится в конфигурационном файле. По умолчанию это файл с именем, как у исполняемого файла, но с расширением `.xml`. Имя файла конфигурации можно задать в командной строке, используя ключ `/C`. Если файла конфигурации нет, то будут использованы минимальные настройки по умолчанию. В случае ошибки в файле конфигурации, при запуске приложения будет показано сообщение об ошибке с номером строки файла, содержащей ошибку.

Минимальный файл конфигурации выглядит следующим образом:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <loodsman auto="1" />
    <workflow auto="1" />
    <mainform />
    <frames auto="1" />
    <plugins auto="1" />
</configuration>
```

Основные элементы в нем:

Элемент	Описание
loodsman	Настройки подключения к серверу приложений и базам данных ЛОЦМАН. Если указано <code>auto="1"</code> , то настройки подключение получаются из реестра, а список баз данных с сервера приложений.
workflow	Настройки подключения к серверу приложений и базам данных Workflow. Если указано <code>auto="1"</code> , то настройки подключение получаются из реестра, а список баз данных с сервера приложений (для версии ЛОЦМАН меньше 10-й это будет одна база данных WorkFlow).
mainform	Настройки основного окна приложения — заголовок окна, возможность настройки меню, место хранения файлов для настраиваемого пользователем меню, место хранения шаблонов ответов.
frames	Список вкладок и их параметры.
plugins	Параметры загрузки плагинов.

### Элемент Loodsman

В атрибуте `sp` могут указываться параметры подключения к серверу приложений. Если атрибута нет, то параметры подключения получаются из реестра. Формат параметров подключения такой же, как и в реестре. Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <loodsman auto="1" sp="AS" />
</configuration>
```

В атрибуте `disconnect` можно задать время простоя до отключения от сервера приложений. Время задается в минутах (0 — запретить автоматическое отключение). По умолчанию 15 минут. Автоматическое отключение от сервера приложений необходимо для экономии лицензий. Для пользователя это полностью прозрачно — подключение к нужной БД и рабочему проекту будет восстановлено автоматически. Например, отключение от сервера приложений при простое более 5 минут:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <loodsman auto="1" disconnect="5" />
</configuration>
```

Запретить отключение от сервера приложений при простое:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <loodsman auto="1" disconnect="0" />
</configuration>
```

Во вложенном элементе `bases` указываются базы данных, с которыми будет работать LWF.

Для каждой базы данных могут указываться параметры подключения и настройки быстрого поиска в этой базе. Обратите внимание, что необходимо указать `auto="0"` для того, чтобы список баз данных читался из файла конфигурации. Список баз:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <loodsman auto="0">
    <bases>
      <base base="ДемоМашиностроение" />
      <base base="Рабочая" />
    </bases>
  </loodsman>
</configuration>
```

Для элемента `base` можно указать атрибуты `username`, `password` и `accessmethod`, с именем пользователя, паролем и способом аутентификации. Способ аутентификации может принимать следующие значения: 0 — Windows; 1 — SQL-сервер, пароль будет запрошен у пользователя; 2 — SQL-сервер, пароль сохранен (и 3 — недоступная база данных, использоваться не должен). Пример параметров подключения:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <loodsman auto="0">
    <bases>
      <base base="ДемоМашиностроение"
        username="semeng"
        password="123456"
        accessmethod="2" />
    </bases>
  </loodsman>
</configuration>
```

Для элемента `base` можно указать настройки быстрого поиска в этой базе, добавив вложенный элемент `search` с атрибутами `types`, `states` и `nameattribute`. Если указать `types`, то быстрый поиск будет выполняться только среди объектов указанных типов. Если указать `states`, то быстрый поиск будет выполняться только среди объектов в указанных состояниях. Если указать `nameattribute`, то быстрый поиск будет выполняться сначала по обозначению (ключевому атрибуту) объекта, и если объектов с искомым обозначением найдено не будет, то будет выполнен поиск по значению указанного в `nameattribute` атрибута. Например, поиск среди объектов с типами «Деталь» или «Сборочная единица» в состояниях «Проектирование» или «Серия», сначала по ключевому атрибуту, затем по атрибуту «Наименование»:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <loadsman auto="0">
    <bases>
      <base base="ДемоМашиностроение">
        <search types="Деталь;Сборочная единица"
              states="Проектирование;Серия"
              nameattribute="Наименование" />
      </base>
    </bases>
  </loadsman>
</configuration>
```

Для элемента `base` можно задать использование локального рабочего диска, добавив атрибут `ls="1"`. Подробнее о локальном рабочем диске написано ниже в главе «Рабочий диск».

Пример включения локального рабочего диска:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <loadsman auto="0">
    <bases>
      <base base="ДемоМашиностроение" ls="1" />
    </bases>
  </loadsman>
</configuration>
```

Атрибут `signfile` у элемента `base` управляет отображением пункта меню «Электронная подпись» для контекстного меню в области файлов. По умолчанию этот пункт меню не показывается. Подробнее об использовании электронной подписи написано ниже в главе «Электронная подпись». Пример включения пункта меню «Электронная подпись»:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <loadsman auto="0">
    <bases>
      <base base="ДемоМашиностроение" signfile="1" />
    </bases>
  </loadsman>
</configuration>
```

Атрибут `syncflags` у элемента `base` задает параметры вызова интегратора при выполнении команды «Получить информацию». Параметры определяют, каким образом информация, полученная из файла, должна повлиять на данные в ЛОЦМАН:PLM. По умолчанию данные изменяются в соответствии с данными из файла.

Атрибут `syncflags` может быть комбинирован из следующих битовых флагов:

Значение	Описание
0	По умолчанию. Данные в ЛОЦМАН:PLM изменяются в соответствии с данными из файла.
1	Не удалять существующие атрибуты. Если в данных из файла отсутствуют атрибуты, которые присутствуют в ЛОЦМАН:PLM, то эти атрибуты из ЛОЦМАН:PLM не удаляются.
2	Не удалять существующие связи. Если в данных из файла отсутствуют связи, которые присутствуют в ЛОЦМАН:PLM, то эти связи из ЛОЦМАН:PLM не удаляются.
4	Зарезервировано. Не используется.
8	Формировать вторичное представление на головной документ.
16	Формировать вторичное представление на все возвращаемые документы.

Таким образом, если задать значение `syncflags="3"`, то есть  $1 + 2$ , то это будет означать выполнение обоих условий «Не удалять существующие атрибуты» и «Не удалять существующие связи».

Пример включения формирования вторичного представления на головной документ:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <lloodsman auto="0">
    <bases>
      <base base="ДемоМашиностроение" syncflags="8" />
    </bases>
  </lloodsman>
</configuration>
```

## Элемент Workflow

В атрибуте `sp` могут указываться параметры подключения к серверу приложений Workflow. Если атрибута нет, то параметры подключения получаются из реестра. Формат параметров подключения такой же, как и в реестре. Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <workflow auto="1" sp="AS" />
</configuration>
```

В атрибуте `execserver` можно указать режим запуска потока для приема соединений от сервера приложений (для выполнения `ExecDllFunction` из скриптов автоопераций). Атрибут может принимать следующие значения: 0 — не запускать поток; 1 — попытаться запустить поток, если получится; 2 — запуск потока обязателен, без него приложение не сможет продолжить работу. Если атрибута нет, будет использован режим 1 — запустить поток, если получится. Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <workflow auto="1" execserver="2" />
</configuration>
```

В атрибуте `wfbusinesslogic` можно включить использование библиотеки `WorkflowBusinessLogic.dll` для движения по бизнес-процессу на стороне клиента и выполнения скриптов автоопераций. Атрибут может принимать следующие значения: 0 — не использовать; 1 — использовать. Если атрибута нет, будет использован режим 0 — не использовать. В зависимости от значения этого атрибута вызовы методов `SetTaskState`, `SetStateOfProcess` и подобных будут либо передаваться серверу приложений, либо в библиотеку `WorkflowBusinessLogic.dll`. При использовании этой библиотеки нет необходимости в запуске потока приема соединений от сервера приложений (библиотека предназначена для замены этого потока), и параметр `execserver` необходимо установить в значение 0. Библиотека входит в состав ЛОЦМАН с версии 2013. Пример:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <workflow auto="1" execserver="0" wfbusinesslogic="1" />
</configuration>
```

В атрибуте `attachdocs` можно разрешить прикрепление документов к процессам и заданиям `Workflow` (связи «Прикреплено к процессу» и «Прикреплено к заданию» в списке связанных объектов). Атрибут может принимать следующие значения: 0 — не разрешено; 1 — разрешено. Если атрибута нет, будет использован режим 1 — разрешено. Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <workflow auto="1" attachdocs="0" />
</configuration>
```

Во вложенном элементе `bases` указываются базы данных (для версии ЛОЦМАН меньше 10-й это будет одна база данных `WorkFlow`). Для каждой базы данных могут указываться параметры подключения. Обратите внимание, что необходимо указать `auto="0"` для того, чтобы список баз данных читался из файла конфигурации. Список баз:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <workflow auto="0">
    <bases>
      <base base="ДемоМашиностроение" />
      <base base="Рабочая" />
    </bases>
  </workflow>
</configuration>
```

Для элемента `base` можно указать атрибуты `username`, `password` и `accessmethod`, с именем пользователя, паролем и способом аутентификации. Способ аутентификации может принимать следующие значения: 0 — Windows; 1 — SQL-сервер, пароль будет запрошен у пользователя; 2 — SQL-сервер, пароль сохранен (и 3 — недоступная база данных, использоваться не должен).

Пример параметров подключения:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <workflow auto="0">
    <bases>
      <base base="ДемоМашиностроение"
        username="semeng"
        password="123456"
        accessmethod="2" />
    </bases>
  </workflow>
</configuration>
```

## Элемент Mainform

В атрибуте `caption` можно указать заголовок окна приложения. В тексте заголовка можно указывать специальные символы: `%F` — имя текущей вкладки и `%B` — имя текущей базы данных. Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <mainform caption="LWF: %F - %B" />
</configuration>
```

В атрибуте `svhost` можно включить открытие модулей просмотра и аннотирования вторичного представления в отдельном процессе. Значения: 0 — выключено, 1 — включено.

Во вложенном элементе `menu` указываются параметры главного меню приложения. В атрибуте `customizable` задается возможность настройки пользователем меню с помощью команды «Вид → Панели → Настроить». Значения: 0 — запретить настройку меню, 1 — разрешить настройку меню. По умолчанию настройка меню запрещена. В атрибуте `default` задается путь к файлу с настройками меню для всех пользователей. По умолчанию это файл с именем, как у исполняемого файла, но с расширением `.menu`. В атрибуте `custom` задается путь к файлу с настройками меню для текущего пользователя. По умолчанию это файл `%APPDATA%\ASCON\LWF\LWF.menu` (для Windows 7 это будет `C:\Users\Имя\AppData\Roaming\ASCON\LWF\LWF.menu`).

Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <mainform>
    <menu customizable="1"
      defaul="%ALLUSERSPROFILE%\ASCON\LWF\Default.menu"
      custom="%LOCALAPPDATA%\ASCON\LWF\Custom.menu" />
  </mainform>
</configuration>
```

Во вложенном элементе `templates` указывается расположение шаблонов ответов. В атрибуте `letter` задается путь к файлу с шаблонами ответов на задания (шаблонами писем) Workflow. По умолчанию это файл `%APPDATA%\ASCON\LWF\Letter.templates` (`C:\Users\Имя\AppData\Roaming\ASCON\LWF\Letter.templates`).

Например:



```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <mainform>
    <templates letter="%APPDATA%\ASCON\LWF\Letters.xml" />
  </mainform>
</configuration>
```

Во вложенном элементе `settings` задаются настройки приложения (команда главного меню «Инструменты → Параметры») по умолчанию и возможность их редактирования. Атрибуты `openfile` и `openurl` управляют настройками «Открывать ссылки на объекты ЛОЦМАН:PLM» и «Открывать гиперссылки на объекты ЛОЦМАН:PLM» соответственно. Атрибуты могут принимать следующие значения: 0 — выключено и недоступно для изменения, 1 — выключено по умолчанию и можно изменить, 2 — включено по умолчанию и можно изменить, 3 — включено и недоступно для изменения. По умолчанию значение атрибутов `openfile` и `openurl` равно 1.

## Элемент Frames

Элемент `frames` задает список вкладок и их параметры. Обратите внимание, что необходимо указать `auto="0"` для того, чтобы список вкладок читался из файла конфигурации. Вкладки будут расположены в том порядке, в котором они следуют в файле конфигурации.

Элемент `frame` описывает вкладку приложения. В атрибуте `type` указывается тип вкладки, доступные типы вкладок будут описаны ниже в таблице. В атрибуте `name` указывается имя вкладки. Имя может содержать символы перевода строки, например "Полученные&#13;&#10;задания". В атрибуте `image` указывается индекс значка вкладки (от 0 до 5).

С помощью атрибута `selected` можно указать вкладку, открываемую при запуске приложения, для этого для нужной вкладки необходимо добавить `selected="1"`. Для некоторых типов вкладок можно указывать дополнительные атрибуты, описанные ниже в таблице.

Вкладки могут быть следующих типов:

Тип	Описание
TaskList	Список полученных заданий.
InitList	Список инициированных процессов.
AuditList	Список контролируемых процессов.
CheckOutTree	Дерево рабочих проектов.
Tree	Дерево объектов. Для элемента <code>frame</code> дополнительно можно указать атрибут <code>simpletree="1"</code> , в этом случае для объектов в дереве не будут получаться дополнительные атрибуты.
FavTree	Избранное.

Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <frames>
    <frame type="TaskList" name="Задания" image="2"
      selected="1" />
    <frame type="InitList" name="Процессы" image="3" />
  </frames>
</configuration>
```

Для элемента `frame` с типом `Tree` необходимо указывать вложенные элементы `projects` и `filters`. Элемент `projects` задает список проектов, которые будут добавлены в дерево. Для того, чтобы добавить в дерево все проекты (из базы данных с именем, как у вкладки), необходимо указать атрибут `auto="1"`. Элемент `filters` в данный момент не используется. Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <frames>
    <frame type="Tree" name="ДемоМашиностроение" image="0">
      <projects auto="1" />
      <filters />
    </frame>
  </frames>
</configuration>
```

В элемент `projects` можно добавить произвольное количество элементов `project`, каждый из которых описывает один проект базы данных, добавляемый в дерево. Для элемента `project` необходимо указать атрибут `base` — имя базы данных и, либо атрибут `versionid` — идентификатор объекта, либо атрибуты `type`, `product` и `version` — соответственно тип, ключевой атрибут и версию объекта.

Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <frames>
    <frame type="Tree" name="Мои проекты" image="0">
      <projects auto="0">
        <project base="ДемоМашиностроение"
          type="Сборочная единица"
          product="078.505.9.0100.00"
          version="1" />
      </projects>
      <filters />
    </frame>
  </frames>
</configuration>
```

## Элемент Plugins

Для элемента `plugins` можно указать атрибут `auto`, задающий загрузку плагинов ЛОЦМАН Клиент приложением LWF. Значения: 1 — загружать плагины, 0 — не загружать плагины. По умолчанию загружаются только плагины, описанные в элементах `plugin`.

Если auto=1, то LWF загружает плагины, список которых содержится в реестре в ветках HKLM\Software\ASCON\Loodsman\Client\PluginManager и HKCU\Software\ASCON\Loodsman\Client\PluginManager, а также в секции PlugIns файла Loodsman.ini (для совместимости со старыми версиями ЛОЦМАН).

Перед загрузкой плагина LWF устанавливает текущим каталогом папку, в которую установлен ЛОЦМАН Клиент (обычно это C:\Program Files (x86)\ASCON\Loodsman\Client). Поэтому если библиотека (DLL) плагина зависит от других библиотек, они должны находиться либо в этой папке, либо в системной папке (C:\Windows\SysWOW64 для 64-х разрядной системы, C:\Windows\System32 для 32-х разрядной системы).

Элемент plugin описывает подключаемый модуль. В атрибуте base указывается имя базы данных, к которой будет подключен модуль, или символ \* (звездочка) для общего для всех баз данных модуля. В атрибуте filename указывается имя файла. Имя файла может быть полным или задано относительно папки, в которую установлен ЛОЦМАН Клиент (обычно это C:\Program Files (x86)\ASCON\Loodsman\Client).

Например:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <plugins auto="0">
    <plugin base="*"
      filename="CalcWeight.pgi" />
    <plugin base="Демо"
      filename="L:\Plugins\CalcFormat.pgi" />
  </plugins>
</configuration>
```

## Рабочий диск

Для изменения файлов приложению требуется рабочий диск. Буква и сетевой путь к рабочему диску настраиваются в конфигураторе ЛОЦМАН:PLM. Приложение автоматически подключит рабочий диск, как только в нем возникнет необходимость.

Для работы с сетевым диском требуется доступ к файловому серверу. В случае, когда нет возможности работать с файловым сервером (например, при подключении к серверу приложений через интернет), можно настроить LWF на работу с локальным рабочим диском. В этом случае буква диска будет той же самой, но диск будет подключен к папке локального диска, и управлять файлами будет не сервер приложений, а клиентское приложение. Диск подключается в папку %APPDATA%\ASCON\LWF\LS\Files (C:\Users\Имя\AppData\Roaming\ASCON\LWF\LS\Files).

Для включения локального рабочего диска необходимо в файле конфигурации в разделе loodsman для нужной базы данных добавить атрибут ls="1". Подробнее об этом написано в главе «Файл конфигурации».

Отключить локальный рабочий диск можно командой subst, например:

```
subst /D L:
```

При использовании локального рабочего диска появятся следующие ограничения:

- Возможно возникновение ошибок при сохранении файлов размером больше 100 Мб, максимально возможный размер файла при этом составит примерно 400 Мб;
- Невозможно добавление вторичного представления и сохранение аннотаций (просмотр вторичного представления и аннотаций будет работать как обычно);
- Не будет работать команда «Получить информацию», так как интегратор поддерживает работу только с сетевым диском.

Если есть возможность использовать сетевой диск, то не рекомендуется использовать локальный рабочий диск.

## Крупный шрифт

LWF использует шрифт, установленный в операционной системе по умолчанию. Изменить размер шрифта по умолчанию можно описанным ниже образом.

Для Windows XP используйте «Панель управления → Оформление и темы → Изменить разрешение экрана → Дополнительно → Общие → Размер шрифта», выберите необходимый размер шрифта: обычный или крупный.

Для Windows 7 используйте «Панель управления → Оформление и персонализация → Экран → Изменение размера текста и других элементов», выберите необходимый размер шрифта: мелкий, средний или крупный.

Кроме того, возможно переопределить используемый по умолчанию размер шрифта только для LWF, для чего в реестре в ветке HKCU\Software\ASCON\LWF нужно создать параметр типа REG\_DWORD с именем `FontSize` и установить его в значение нужного размера шрифта: 10 — средний, 12 — крупный.

## Особенности при работе на терминальном сервере

Для выполнения функции `ExecDllFunction` из автоопераций Workflow приложение создает специальный поток, который слушает порт 8090 и принимает соединения от сервера приложений.

При работе на терминальном сервере такой поток создаваться не будет (так как невозможно создание нескольких таких потоков на одном компьютере) и функции `ExecDllFunction` в автооперациях Workflow работать не будут.

При работе в Windows Virtual PC с включенными компонентами интеграции приложение также запускается в терминальном режиме. Для того, чтобы поток приема соединений запускался в этом случае, необходимо в файле конфигурации в разделе `workflow` добавить атрибут `execserver="2"`. Подробнее об этом написано в главе «Файл конфигурации».

## Электронная подпись

LWF позволяет работать с электронной подписью для объектов и файлов таким же образом, как это сделано в ЛОЦМАН:PLM.

В зависимости от версии сервера приложений выбирается один из двух вариантов электронной подписи — как в версиях 8.5–2013 или как в версии 2014. Эти варианты электронной подписи не совместимы между собой.

При использовании электронной подписи в ЛОЦМАН:PLM версий 8.5–2013 следует учесть, что она сделана не совсем правильно. Например, не проверяется цепочка доверия для сертификатов, поэтому правильность подписи можно проверить только вручную, сверяя отпечаток сертификата с известным заранее. Кроме того, подпись файла нельзя сохранить отдельно и передать третьей стороне для проверки подлинности файла.

LWF поддерживает только усиленную электронную подпись. Усиленная подпись требует наличия специального сертификата, который выдается удостоверяющим центром. Выданные сертификаты хранятся на компьютере в хранилище сертификатов Windows.

Для работы с электронной подписью объектов необходимо вынести пункт меню «Электронная подпись» в главное меню, на панель инструментов или в контекстное меню нужных вкладок. Для этого перейдите в режим настройки меню и панелей инструментов, выбрав в меню «Вид → Вкладки → Настроить» (команда доступна, только если это разрешено в файле конфигурации приложения). Перетащите из списка команд категории «Инструменты» команду «Электронная подпись» в нужное вам место. Нажмите «Закрыть» и согласитесь с сохранением настроек на диск. Для того, чтобы команда была доступна всем пользователям, подготовьте общий для всех файл LWF.menu.

Для работы с электронной подписью файлов необходимо в файле конфигурации в разделе loodsman для нужной базы данных добавить атрибут `signfile="1"`. Подробнее об этом написано в главе «Файл конфигурации».

## Используемые лицензии

Система ЛОЦМАН:PLM состоит из трех основных компонентов: базы данных, сервера приложений и клиента. Лицензирование реализовано разработчиком (ЗАО «АСКОН») на сервере приложений — при подключении клиента сервер приложений запрашивает лицензию у сервера лицензий. LWF заменяет собой только стандартный клиент, и поэтому ему необходима лицензия на подключение к серверу приложений.

Есть два типа лицензий:

- loodsmanclient: запуск ЛОЦМАН Клиент;
- loodsmangeneric: доступ к серверу приложений.

Посмотреть доступные вам лицензии и их количество вы можете с помощью программы LMTOOLS (обычно расположена в папке `C:\Program Files (x86)\Common Files\ASCON Shared\FLEXlm`).

LWF использует лицензию loodsmangeneric, но если все такие лицензии заняты, то будет использована лицензия loodsmanclient.

Лицензии loodsmangeneric стоят примерно в два раза дешевле, чем лицензии loodsmanclient. Лицензии приобретаются у ЗАО «АСКОН».