

УВЕРЕННОСТЬ В КАЖДОМ РЕШЕНИИ.



Компания «Гарант»
Система ГАРАНТ Платформа F1 ЭКСПЕРТ

ГАРАНТ — совместимо!

Описание
библиотеки
интеграции

2010 г.

Оглавление

ВВЕДЕНИЕ	3
ФОРМАТ КОМАНД И ССЫЛОК	3
ОПИСАНИЕ ФУНКЦИЙ	4
ФУНКЦИЯ IsGARANTINSTALLED	4
ФУНКЦИЯ GARANTPROCESSCOMMAND	4
ФУНКЦИЯ GARANTDEMOPROCESSCOMMAND	5
ФУНКЦИЯ GARANTSHOWLINK	5
ФУНКЦИЯ GARANTDEMO SHOWLINK	6
Значения кодов ошибок, возвращаемых функциями	7
СПОСОБ ПОЛУЧЕНИЯ ССЫЛОК	7
СПОСОБ ЗАДАНИЯ КЛЮЧЕВОЙ ФРАЗЫ	7
ПРИЛОЖЕНИЕ 1	8
Список команд	8
Формат XML-ССЫЛКИ	8
ПРИМЕРЫ XML-ССЫЛОК	12

Введение

Библиотека интеграции предназначена для облегчения процесса интеграции программ сторонних производителей с системой ГАРАНТ Платформа F1 ЭКСПЕРТ.

Библиотека предоставляет разработчику продукта, интегрирующегося с системой ГАРАНТ сервис, включающий в себя функции для запуска системы ГАРАНТ и передачи ей команд на поиск и/или показ документов.

Взаимодействие сторонней программы с системой ГАРАНТ осуществляется с помощью команд и ссылок. Команда – предназначена для активации в системе ГАРАНТ различных окон в интерфейсе пользователя (основанное меню, правовой навигатор, справочная информация и пр.). Ссылка – предназначена для открытия объектов системы, разрешенных к сохранению (списки, запросы и пр.). Хранение команд и ссылок является задачей сторонней программы.

Формат команд и ссылок

Команды представляют собой целочисленные идентификаторы, предназначенные для активации определенного окна в интерфейсе пользователя. Список допустимых команд приводится в Приложении 1.

Предусматривается возможность переключения на следующие окна:

1. Основное меню
2. Поиски (по ситуации, по реквизитам, по источнику опубликования, ПРАЙМ. Обзор изменений законодательства)
3. Правовой навигатор
4. Толковый словарь
5. Справочная информация\новые документы
6. Просто запустить/активировать текущее окно

Ссылки представляют собой xml-текст, предназначенный для описания объектов системы, разрешенных к сохранению. Формат xml-текста приводится в Приложении 1.

Предусматривается возможность ссылаться на следующие типы объектов:

1. Закладка на документ (как на параграф, так и на SUB/Block).
2. Запрос на поиск (по ситуации, по реквизитам, по источнику опубликования с правовой поддержкой, обзор изменений законодательства) – при открытии автоматически выполняется поиск по сохраненным параметрам и показывается его результат (Список для всех поисков, кроме ПРАЙМ. Обзора изменений законодательства).
3. Список.

Описание функций

Функция IsGarantInstalled

```
function IsGarantInstalled: integer; stdcall;  
int stdcall IsGarantInstalled ();
```

Функция выполняет проверку наличия на машине проинсталлированной системы ГАРАНТ (без проверки ее работоспособности).

Возвращаемые значения: функция возвращает GI_OK в случае, если система ГАРАНТ проинсталлирована и GI_NOTFOUND в противном случае.

Примечание: В реестре ищется ключ

HKEY_CURRENT_USER\Software\Garant.F1\ApplicationAdapterOptions и по его значению вычисляется путь к исполняемому модулю системы ГАРАНТ. После чего проверяется наличие исполняемого файла системы ГАРАНТ по этому пути.

Функция GarantProcessCommand

```
function GarantProcessCommand(aCommand: DWORD;  
OpenInNewWindow: Bool; aTimeOut: DWORD):integer;  
stdcall;
```

```
int stdcall GarantProcessCommand(DWORD aCommand, bool
    OpenInNewWindow, DWORD aTimeOut);
```

Функция принимает команду и пытается передать ее системе ГАРАНТ для обработки. При этом функция возвращает управление, не дожидаясь окончания обработки ссылки, системой ГАРАНТ.

Параметры:

- aCommand – команда.
- OpenInNewWindow – если система ГАРАНТ открыта, нужно ли открывать для обработки новое главное окно или обрабатывать ссылку в последнем открытом главном окне системы ГАРАНТ.
- aTimeOut – тайм-аут для ожидания отклика системы ГАРАНТ, в миллисекундах (по аналогии с функциями синхронизации, в т.ч. с обработкой INFINITE).

Возвращаемые значения: функция возвращает GI_OK в случае, если система ГАРАНТ успешно обработала запрос или значения ошибок GI_INVALIDLINKFORMAT, GI_BUSY, GI_TIMEOUT или GI_NOTFOUND.

Примечание: В случае если система ГАРАНТ еще не запущена, в реестре ищется ключ HKEY_CURRENT_USER\Software\Garant.F1\ApplicationAdapterOptions и по его значению вычисляется путь к исполняемому модулю системы ГАРАНТ.

Авторизация при первом запуске системы ГАРАНТ проходит обычным образом.

Функция GarantDemoProcessCommand

```
function GarantDemoProcessCommand(aPath: PChar;
    anIntegrationKey: PChar; aCommand: DWORD;
    OpenInNewWindow: Bool; aTimeOut: DWORD):integer;
    stdcall;
```

```
int stdcall GarantDemoProcessCommand(char* aPath, char*
    anIntegrationKey, DWORD aCommand, bool
    OpenInNewWindow, DWORD aTimeOut);
```

Функция принимает команду и пытается передать ее системе ГАРАНТ для обработки. При этом функция возвращает управление, не дожидаясь окончания обработки ссылки, системой ГАРАНТ.

Параметры:

- aPath – путь, по которому ищется исполняемый модуль системы ГАРАНТ.
- anIntegrationKey – ключевая фраза для проверки того, что запущенная система ГАРАНТ подходит. Описание и способ задания ключевой фразы приводиться ниже.
- aCommand – команда.
- OpenInNewWindow – если система ГАРАНТ открыта, нужно ли открывать для обработки новое главное окно или обрабатывать ссылку в последнем открытом главном окне системы ГАРАНТ.
- aTimeOut – тайм-аут для ожидания отклика системы ГАРАНТ, в миллисекундах (по аналогии с функциями синхронизации, в т.ч. с обработкой INFINITE).

Возвращаемые значения: функция возвращает GI_OK в случае, если система ГАРАНТ успешно обработала запрос или значения ошибок GI_INVALIDLINKFORMAT, GI_ALREADYSTARTED, GI_BUSY, GI_TIMEOUT или GI_NOTFOUND.

Примечание: Одно из применений функции – запуск системы ГАРАНТ с демонстрационного диска.

В случае если проверка ключевой фразы окончилась неудачно, функция возвращает ошибку GI_ALREADYRUNNING. Проверка проводится всегда, даже если система ГАРАНТ еще не была запущена.

Авторизация при первом запуске системы ГАРАНТ проходит обычным образом.

Функция GarantShowLink

```
function GarantShowLink(aRef:PChar; OpenInNewWindow: Bool;
    aTimeOut: DWORD):integer; stdcall;
int stdcall GarantShowLink(char * aRef, bool
    OpenInNewWindow, DWORD aTimeOut);
```

Функция принимает ссылку и пытается передать ее системе ГАРАНТ для обработки. При этом функция возвращает управление, не дожидаясь окончания обработки ссылки, системой ГАРАНТ.

Если для ссылки на документ не удастся найти соответствующий параграф/саб, то документ позиционируется на начало (без вывода каких либо сообщений об ошибках).

Если для списка отсутствует часть документов в текущей базе, то список открывается без этих документов (без вывода каких-либо сообщений об ошибках).

Параметры:

- aRef – ссылка в любом из допустимых форматов.
- OpenInNewWindow – если система ГАРАНТ открыта, нужно ли открывать для обработки новое главное окно или обрабатывать ссылку в последнем открытом главном окне системы ГАРАНТ.
- aTimeout – тайм-аут для ожидания отклика системы ГАРАНТ, в миллисекундах (по аналогии с функциями синхронизации, в т.ч. с обработкой INFINITE).

Возвращаемые значения: функция возвращает GI_OK в случае, если система ГАРАНТ успешно обработала запрос или значения ошибок GI_INVALIDLINKFORMAT, GI_BUSY, GI_TIMEOUT, GI_SYSTEMERROR или GI_NOTFOUND.

Примечание: В случае если система ГАРАНТ еще не запущена, в реестре ищется ключ HKEY_CURRENT_USER\Software\Garant.F1\ApplicationAdapterOptions и по его значению вычисляется путь к исполняемому модулю системы ГАРАНТ.

Авторизация при первом запуске системы ГАРАНТ проходит обычным образом.

Функция GarantDemoShowLink

```
function GarantDemoShowLink(aPath: PChar; anIntegrationKey:
PChar; aRef:PChar; OpenInNewWindow: Bool; aTimeout:
DWORD):integer; stdcall;
int stdcall GarantDemoShowLink(char* aPath, char*
anIntegrationKey, char* aRef, bool OpenInNewWindow,
DWORD aTimeout);
```

Функция принимает ссылку и пытается передать ее системе ГАРАНТ для обработки. При этом функция возвращает управление, не дожидаясь окончания обработки ссылки, системой ГАРАНТ.

Если для ссылки на документ не удастся найти соответствующий параграф/саб, то документ позиционируется на начало (без вывода каких либо сообщений об ошибках).

Если для списка отсутствует часть документов в текущей базе, то список открывается без этих документов (без вывода каких-либо сообщений об ошибках).

Параметры:

- aPath – путь, по которому ищется исполняемый модуль системы ГАРАНТ.
- anIntegrationKey – ключевая фраза для проверки того, что запущенная система ГАРАНТ подходит. Описание и способ задания ключевой фразы приводиться ниже.
- aRef – ссылка в любом из допустимых форматов.
- OpenInNewWindow – если система ГАРАНТ открыта нужно ли открывать для обработки новое главное окно или обрабатывать ссылку в последнем открытом главном окне системы ГАРАНТ.
- aTimeout – тайм-аут для ожидания отклика системы ГАРАНТ, в миллисекундах (по аналогии с функциями синхронизации, в т.ч. с обработкой INFINITE).

Возвращаемые значения: функция возвращает GI_OK в случае, если система ГАРАНТ успешно обработала запрос или значения ошибок GI_INVALIDLINKFORMAT, GI_ALREADYSTARTED, GI_BUSY, GI_TIMEOUT, GI_SYSTEMERROR или GI_NOTFOUND.

Примечание: Одно из применений функции – запуск системы ГАРАНТ с демонстрационного диска.

В случае если проверка ключевой фразы окончилась неудачно, функция возвращает ошибку GI_ALREADYRUNNING. Проверка проводится всегда, даже если система ГАРАНТ еще не была запущена.

Авторизация при первом запуске системы ГАРАНТ проходит обычным образом.

Значения кодов ошибок, возвращаемых функциями

GI_OK = 0 // Нет ошибок

GI_NOTFOUND = 1 // Система ГАРАНТ не найдена

GI_INVALIDLINKFORMAT = 2 // Неверный формат ссылки

GI_BUSY = 3 // Система ГАРАНТ занята (отображает модальный диалог, печатает и т.д.) и не может обработать запрос
GI_TIMEOUT = 4 // Тайм-аут при получении ссылки на главное окно системы ГАРАНТ.
GI_ALREADYRUNNING = 5 // Система ГАРАНТ уже запущена
GI_SYSTEMERROR = 6 // Общесистемная ошибка (нет xml-парсера, битая база и пр.)

Способ получения ссылок

Для получения ссылок на объекты системы, разрешенные к сохранению, можно использовать операцию «Экспорт для интеграции» в контекстном меню системы ГАРАНТ для окна Мои документы, которая будет копировать в буфер обмена сформированную ссылку.

Операция доступна только для некоммерческой версии!

Способ задания ключевой фразы

Ключевая фраза представляет собой произвольный текст, задаваемый пользователем. Определяется она в серверном ini-файле, в разделе [F1Server Params], в ключе с именем «-IntegrationComplectID».

Полное имя серверного ini-файла записано в ключе

HKKEY_LOCAL_MACHINE\Software\Garant.F1\ApplicationServerOptions реестра.

Приложение 1

Список команд

GC_ACTIVATE = 0 // Просто запустить/активировать текущее окно
GC_MAIN_MENU = 1 // Основное меню
GC_NAVIGATOR = 2 // Правовой навигатор
GC_SITUATION_SEARCH = 3 // Поиски по ситуации,
GC_ATTRIBUTES_SEARCH = 4 // Поиски по реквизитам,
GC_PUBLISH_SOURCE_SEARCH = 5 // Поиски по источнику опубликования,
GC_LAW_SUPPORT_SEARCH = 6 // Поиски с правовой поддержкой
GC_REVIEW = 7 // Обзор изменений в законодательстве
GC_DICTION = 8 // Толковый словарь
GC_NEW_DOCS = 9 // Новые документы из Справочной информации

Формат xml-ссылки

Поскольку предусмотрен механизм получения ссылок средствами оболочки, формат приводится исключительно в справочных целях и может быть безболезненно изменен в любой момент.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
```

```
<<Данные ссылки>>
```

Раздел <<Данные ссылки>> содержит данные либо для закладок (bookmark), либо для списков (list), либо для запросов (query).

Ниже приводится DTD для тэгов bookmark, list и query.

```
<!-- Закладки -->
```

```
<!ELEMENT bookmark EMPTY>
```

```
<!ATTLIST bookmark
```

```
xml_version CDATA #REQUIRED
```

```
<!-- текущая версия формата для закладок, на сегодня - "1" -->
```

```
id CDATA #REQUIRED
```

```
<!-- id документа -->
```

```
point CDATA #REQUIRED
```

```
<!-- идентификатор вхождения -->
```

```
point_type (SUB,PAPA) #REQUIRED
```

```
<!-- тип вхождения -->
```

```
>
```

```
<!-- Список -->
```

```
<!ELEMENT list (document*)>
```

```

<!ATTLIST list
xml_version CDATA #REQUIRED
<!-- текущая версия формата для списка, на сегодня - "1" -->
name CDATA #REQUIRED
<!-- имя списка -->
sort_order (descending, ascending) #REQUIRED
<!-- порядок сортировки -->

sort_type (priority, create_date, last_edit_date, not_sorted,
relevance) #REQUIRED
<!-- тип сортировки
st_priority - по юридической силе,
st_create_date - по дате создания,
st_last_edit_date - по дате последнего изменения,
st_not_sorted - в .натуральном. порядке,
st_relevance - по релевантности-->
>
<!-- Элемент списка -->
<!ELEMENT document EMPTY>
<!ATTLIST document
id CDATA #REQUIRED
<!-- id документа -->
point CDATA #REQUIRED
<!-- идентификатор вхождения -->
point_type (SUB,PAPA) #REQUIRED
<!-- тип вхождения -->
>
<!-- Запросы -->
<!ELEMENT query (index_data)+>
<!ATTLIST query
xml_version CDATA #REQUIRED
<!-- текущая версия формата для запросов, на сегодня - "1" -->
type (QT_ATTRIBUTE, QT_OLD_ATTRIBUTE, QT_KW,
QT_PUBLISHED_SOURCE, QT_REVIEW, QT_CONSULT) #REQUIRED
<!-- тип запроса
QT_ATTRIBUTE - поиск по реквизитам,
QT_OLD_ATTRIBUTE - поиск по реквизитам (версия 5.x)
QT_KW - поиск по ситуации,
QT_PUBLISHED_SOURCE - поиск по источнику опубликования,
QT_REVIEW - обзор изменения законодательства,
QT_CONSULT - поиск с правовой поддержкой -->
>
<!-- Элемент запроса -->
<!ELEMENT index_data (logic_group+)>
<!ATTLIST index_data
tag (AT_CONSULTATION_TEXT, AT_TEXT_BODY, AT_TEXT_NAME, AT_TYPE,
AT_SOURCE, AT_CLASS, AT_PUBLISHING_DATE, AT_REG_NUM, AT_REG_DATE,
AT_REG_IN_MU_NUM, AT_REG_IN_MU, AT_NORM, AT_STATUS, AT_EDITION,
AT_PREFIX, AT_TERRITORY, AT_BASES, AT_PUBLISH_SOURCE, AT_KW,
AT_TEXT
)#REQUIRED
<!-- тэг, идентифицирующий что ищем.
AT_CONSULTATION_TEXT
AT_TEXT_BODY
AT_TEXT_NAME
AT_TYPE
AT_SOURCE
AT_CLASS
AT_PUBLISHING_DATE
AT_REG_NUM

```

AT_REG_DATE
AT_REG_IN_MU_NUM
AT_REG_IN_MU
AT_NORM
AT_STATUS
AT_EDITION

AT_PREFIX
AT_TERRITORY
AT_BASES
AT_PUBLISH_SOURCE
AT_KW
AT_TEXT

Допустимость значения от значения атрибута query.tag
(справочно) приводится ниже. -->

type (date, attribute, attnumber, context, string, none)

#REQUIRED

<!-- тип данных

date - дата,

attribute - атрибут,

attnumber - атрибут с выпадающим деревом,

context - контекст поиска, строка,

string - просто строка,

none - -->

>

<!ELEMENT logic_group (full_name+ | (from?, to?))>

<!-- наличие тэгов зависит от index_data.type: для значения

“date” допустимы только from и to (причем один из них обязательно
должен присутствовать, для всех остальных - только full_name)

У одного index_data не может быть двух logic_group с одинаковыми
значениями атрибута value -->

<!ATTLIST logic_operation

value (LO_AND, LO_OR, LO_NOT) #REQUIRED

>

<!-- Собственно искомое значение -->

<!ELEMENT full_name #PCDATA>

<!ELEMENT from EMPTY>

<!ATTLIST from

day CDATA #REQUIRED

month CDATA #REQUIRED

year CDATA #REQUIRED

>

<!ELEMENT to EMPTY>

<!ATTLIST to

day CDATA #REQUIRED

month CDATA #REQUIRED

year CDATA #REQUIRED

>

Допустимость значения атрибута index_data.tag и значение атрибута index_data.type от значения атрибута query.type

index_data.tag	index_data.type	query.type				
		QT_ATT RIBUTE	QT_OL D_ATT RIBUTE	QT_KW	QT_PUB LISHED_ SOURCE	QT_RE VIEW
AT CONSULTATION_TEXT	string					
AT TEXT BODY	context	+				
AT TEXT NAME	context	+				
AT TYPE	attnumber	+	+			
AT SOURCE	attnumber	+	+			+
AT CLASS	attnumber	+	+			+
AT PUBLISHING DATE	date	+	+			+
AT REG NUM	attnumber	+	+			
AT REG DATE	date	+	+			
AT REG IN MU_NUM	attnumber	+	+			
AT REG IN MU	attnumber	+				
AT NORM	attnumber	+				
AT STATUS	attnumber	+				
AT EDITION	attnumber	+				
AT PREFIX	attnumber	+				+
AT TERRITORY	attnumber	+				
AT BASES	attnumber	+				
AT PUBLISH SOURCE	attnumber				+	
AT KW	attnumber			+		
AT TEXT	context		+			

Примеры XML-ссылок

