GFI FaxMaker™

API MANUAL

Shows you how to programmatically send faxes from other applications using the GFI FaxMaker APIs.

Document Version: 20.0

Last updated (month/day/year): 02/02/2021

Contents

1 GFI FaxMaker APIs

With APIs you can integrate your third party software with GFI FaxMaker.

This process automation can be used for a wide variety of industries and applications, such as healthcare, retail and banking.

For example, use XMLAPI with your invoicing software to create an XML report template for account statements and balances. Periodically, generate this report for each client that owes money and store it to the XMLAPI pickup folder. GFI FaxMaker automatically transmits the statements to their intended recipients.

Types of APIs available in GFI FaxMaker:

XMLAPI	Send faxes or SMS using XML files. You can configure third party software to generate APIs automatically and transmit faxes in bulk. GFI FaxMaker picks up and transmits all xml files stored in a custom XMLAPI pickup folder. XMLAPIs must be formatted to a standard convention, using fields and parameters which GFI FaxMaker can use to process faxes and SMS. For more information, refer to XMLAPI (page 12).
TextAPI	Send faxes using plain text files. You can configure third party software to generate APIs automatically and transmit faxes in bulk. GFI FaxMaker picks up and transmits all text files stored in a custom TextAPI pickup folder. TextAPIs must be formatted to a standard convention, using fields and parameters which GFI FaxMaker can use to process faxes and SMS. For more information, refer to TextAPI (page 6).
Web Service API	The GFI FaxMaker Web Service API enables you to develop your own applications that can connect directly with GFI FaxMaker. You can then send and receive faxes directly from your application. For more information, refer to Web Service API (page 26).

2 TextAPI

TextAPI is a feature in GFI FaxMaker that allows transmission of faxes from text files. These text files must be formatted to a standard convention, using fields and parameters which GFI FaxMaker can use to transmit faxes. Store TextAPIs in a dedicated folder. GFI FaxMaker picks up all TextAPIs stored in this folder and transmits content via fax.

To configure TextAPI:

» Create and dedicate a folder that is accessible by GFI FaxMaker where text files for fax transmission will be stored.

» Enable TextAPI feature from GFI FaxMaker. For more information, refer to Enabling TextAPI in GFI FaxMaker (page 6).

» Create TextAPIs using fields and parameter accepted by GFI FaxMaker. For more information, refer to Creating TextAPIs (page 7).

2.1 How TextAPI works

Screenshot 1: How TextAPI works

Step	Description
1	**Generate TextAPI** Usually a 3rd party application is configured to automatically generate TextAPIs. Ensure that the generated file complies to the parameters, fields and requirements of GFI FaxMaker, as described in this manual. For more information, refer to Creating TextAPIs (page 7).
2	**Store the API in the TextAPI pickup folder** In GFI FaxMaker, configure the folders from where to pick up APIs from. Store generated API in the appropriate folder. For more information, refer to Enabling TextAPI in GFI FaxMaker (page 6).
3	**GFI FaxMaker retrieves API** Periodically, GFI FaxMaker polls configured folders for *.txt APIs.
4	**GFI FaxMaker processes and transmits fax** GFI FaxMaker processes the retrieved APIs and transmits the fax according to the fields and parameters specified in the API.

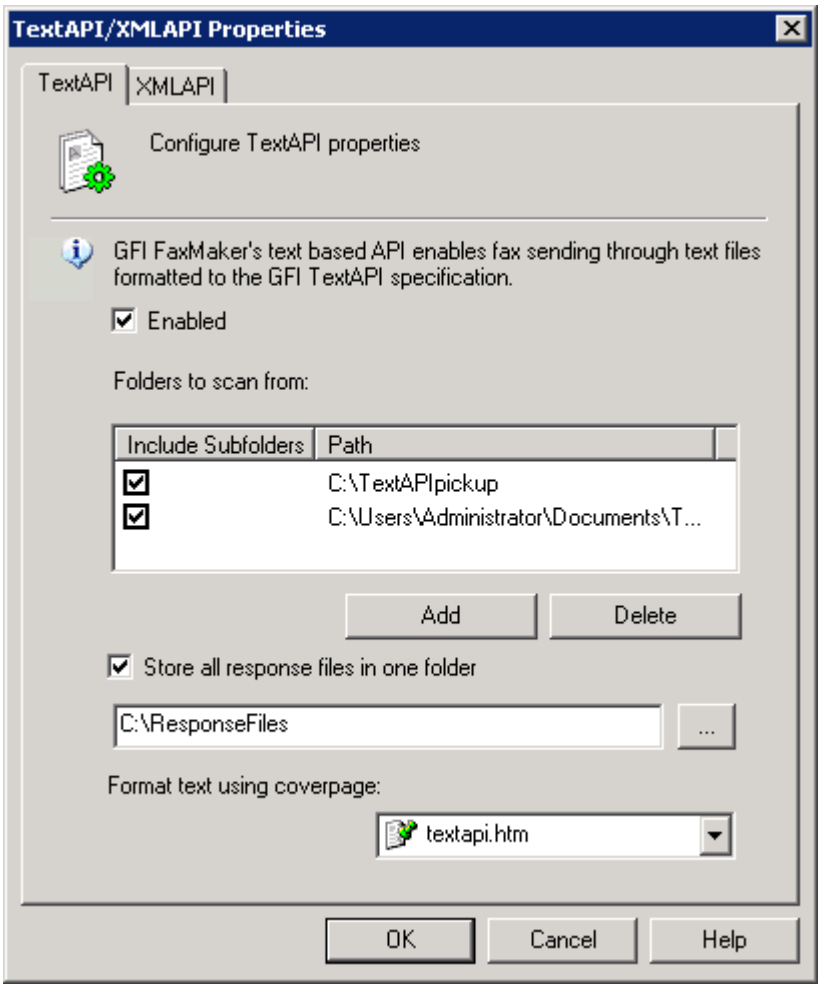
NOTE

Ensure that attachments and body files are fully stored in the appropriate location before storing APIs in pickup folder. Store any attachments in the TextAPI folder.

2.2 Enabling TextAPI in GFI FaxMaker

To enable and configure TextAPI:

1. From GFI FaxMaker Configuration, right-click **Advanced > Text-API/XMLAPI** node and select **Properties**.



Screenshot 2: Enabling TextAPI

2. From **TextAPI** tab, configure the following options:

Option	Description
Enabled	Enables TextAPI.
Folders to scan	Specify the folders where Text APIs will be stored for GFI FaxMaker to process. Click **Add** to select folders. To run recursive TextAPIs scanning on sub-folders, select **Include Subfolders**.
Store all response files in one folder	After processing a TextAPI which does not have the FROM parameter specified, GFI FaxMaker creates a response file. By default, GFI FaxMaker stores response files in the pickup folders. To store all response files in one location, select this option and specify a location. For more information, refer to TextAPI response files (page 10).
Format text using coverpage	Select the coverpage to use for TextAPI faxes. For more information about coverpages, refer to the GFI FaxMaker Administration & Configuration Manual.

3. Click **OK**

Next step: Start creating TextAPIs. For more information, refer to Creating TextAPIs (page 7).

2.3 Creating TextAPIs

When using TextAPI, create a text file (.txt format) that is structured as follows:

Line	Description

First line	Specify the recipient's fax number preceded by a double colon, in the following format:
	`::fax number`
	For example:
	`::12345678`
	Other parameters can also be included in the following format:
	`::first name, company, last name, department, email address, fax number`
	For example, to send a fax to John Smith who is in Sales department at company John Company and whose email address and fax number are `sales@johncompany.com` and `12345678` respectively, the entry will be as follows:
	`::John, John Company, Smith, Sales, sales@johncompany.com, 12345678`
	Note:
	First name should start with an alphabetical character.

Second line (optional)	Specify other sending options preceded by a double colon. For example:
	`::S=pricelist`
	Separate multiple parameters by commas. For example:
	`::c=cover1,23:15,B=5`
	For more information, refer to TextAPI sending options (page 8).

Other lines	Enter the message text to fax.

2.3.1 TextAPI sending options

A number of sending options can be specified in the second line of the text API.

NOTE

Multiple options can be specified. Separate multiple parameters by commas. For example:

`::s=subject,from=Peter,c=coverpage1,23:15,B=5`

NOTE

GFI FaxMaker is not case-sensitive. Parameters can be specified in both capital or small letters.

Sending option	Parameter	Description

Subject	`::S=This is the subject`
	or
	`::subject=This is the subject`
		Specify the subject of the fax. This text is inserted in the `<subject>` field of the coverpage. Replace `This is the subject` with the subject text.

Sender	`::fr=name`
	or
	`::from=name`
		Use this parameter to specify the name of the fax sender. The name specified is included as the sender's name in the coverpage. Replace `name` with the sender's name.
		NOTE: When this option is not specified, GFI FaxMaker has no information about the sender and therefore cannot return a transmission report via email. Instead a response file is generated. For more information, refer to TextAPI response files (page 10).

Sending option	Parameter	Description
Attachment	`::A=FileName`	Specify a file to attach to the fax. The file must be stored in the same TextAPI pickup folder. Replace `FileName` with the name and extension of the file to attach. For example, `::A=FaxAttachment.tif` Multiple files can be attached by specifying the command multiple times. For example: `::A=file1.tif,A=file2.tif` **NOTE:** Attachments are permanently removed from the pickup folder. If a copy is required, ensure that backup is taken before moving attachment to the TextAPI folder.

Priority	**High priority**
	`::p=high`
	or
	`::p=h`
	or
	`::priority=high`
	or
	`::priority=h`
	Low priority
	`::p=low`
	or
	`::p=l`
	or
	`::priority=low`
	or
	`::priority=l`
		Specify the fax priority (low or high) to use when sending the fax.

Billing code	`::B=nnnn`
	or
	`::billingcode=nnnn`
		Use this parameter to specify a billing code, if GFI FaxMaker is configured to require a billing code. Replace `nnnn` with the billing code.

Schedule	`::YYYY-MM-DD-hh:mm:ss`
	or
	`::hh:mm`
		Specify the date and time, or time only when to send the fax.
		Replace:
		» YYYY - year
		» MM - month
		» DD - day
		» hh - hours in 24-hour clock format
		» mm - minutes
		» ss - seconds
		For example, `::2021-04-21-16:05:00`

Coverpage	`::C=CoverpageName`
	or
	`::Coverpage=CoverpageName`
		By default, GFI FaxMaker uses the default coverpage configured in GFI FaxMaker. Use this parameter to send fax using a particular coverpage.
		Replace `CoverpageName` with the name of the coverpage to use, as configured in GFI FaxMaker.
		You can also use `::C=none` to not add a coverpage.

Front covernote	`::F=FrontCovernote`
	or
	`::frontcover=FrontCovernote`
		Use this parameter to send fax using a particular front covernote. Replace `FrontCovernote` with the name of the front covernote to use, as configured in GFI FaxMaker.

Sending option	Parameter	Description

Fax line	`::line=n`
	or
	`::l=n`
		Specify this parameter to request the fax to be sent on a particular fax line. Replace `n` with the fax line number as configured in GFI FaxMaker Configuration.
		NOTE: If the line is busy or unavailable, the fax is sent on another line. To send the fax strictly from a particular line, use the following parameter: `::dl=n`
		The fax is not sent until this fax line is available.
		NOTE: `dl` parameter is also available when using TextAPI commands in SMS to request an SMS to be sent over particular GSM line.

Resolution	**High resolution**
	`::H`
	or
	`::high`
	Normal resolution
	`::N`
	or
	`::normal`
		Use this parameter to specify the resolution of the fax. Use high (200 * 200 dpi) or normal (200 * 100 dpi) resolution.

Header	`::fh=header`
	or
	`::faxheader=header`
		Use this option to set a custom personal header that is added to the default fax header at the top of the fax. Replace `header` with the header text to add.

Transmission report summary	Send transmission report summary
	`::tr=summary`
	Never send summary
	`::tr=none`
		Use `summary` to receive only one transmission report when sending a fax to multiple recipients.
		Use `none` to not send a transmission report after fax transmission.

2.3.2 TextAPI example

```
::John, John's Company, Smith, Sales, sales@johnscompany.com, 12345678
::from=Bob Jones,subject=Hi John,H,fh=Test fax
This is the fax message.
Best regards,
Paul
```

2.3.3 TextAPI response files

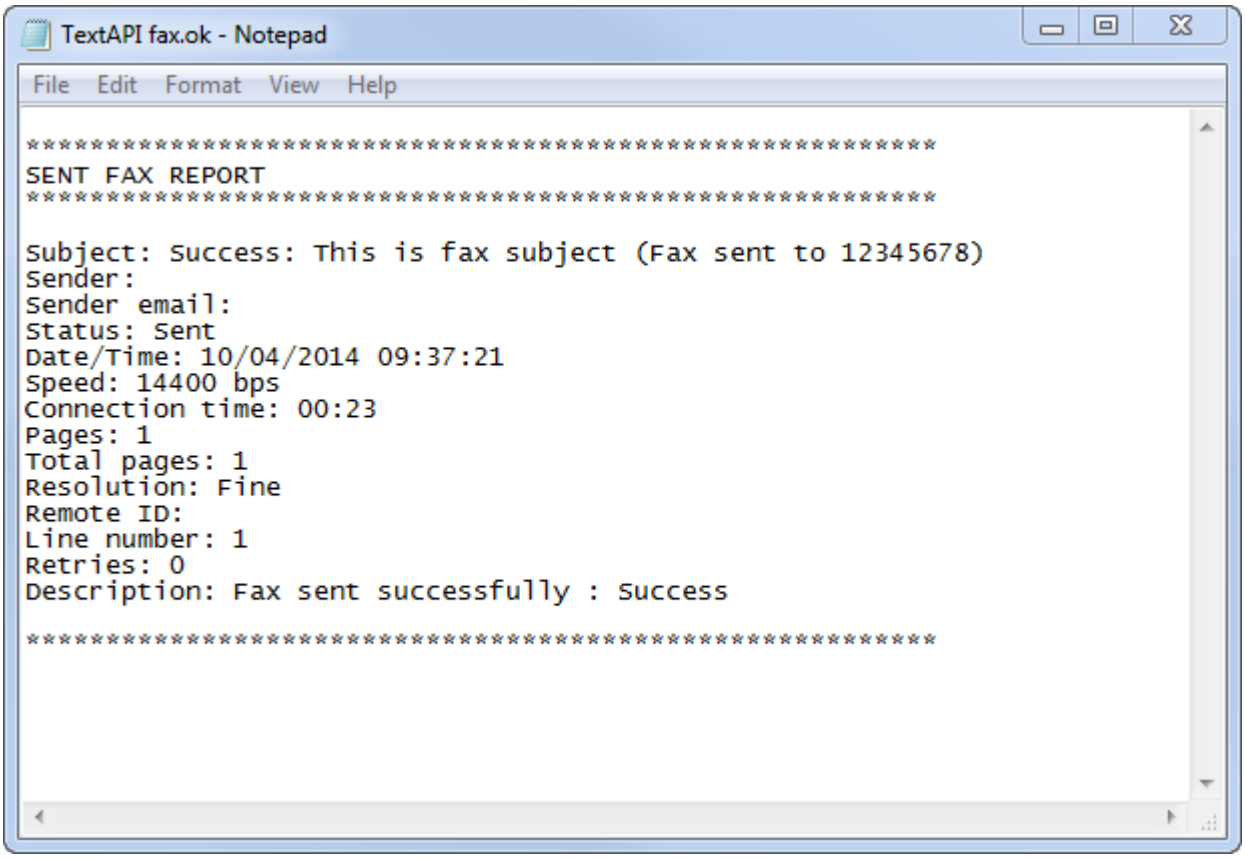
After processing a TextAPI that does not have the sender specified (`from` or `fr` options in the second line), GFI FaxMaker creates a response file to indicate the fax transmission result. If the sender is specified, the response message is sent via email to the sender as a transmission report.

By default the response file is created in the same folder from where the TextAPI was picked up. Alternatively GFI FaxMaker can be configured to store all response files in one location. For more information, refer to Store all response files in one folder (page 7).

The name of the response file matches the name of the TextAPI. The extension of the file depends on the transmission result:

» `*.ok` - indicates that the fax was transmitted successfully.

» `*.err` - indicates that the fax could not be sent. Review the **Description** message to help you with troubleshooting.



Screenshot 3: Sample TextAPI response message

3 XMLAPI

XMLAPI is a feature in GFI FaxMaker that allows transmission of faxes from XML files. These XML files must be formatted to a standard convention, using fields and parameters which GFI FaxMaker can use to transmit faxes. Store XMLAPIs in a dedicated folder. GFI FaxMaker picks up all XMLAPIs stored in this folder and transmits content via fax.

To configure XMLAPI:

» Create and dedicate a folder that is accessible by GFI FaxMaker where XML files for fax transmission will be stored.

» Enable XMLAPI feature from GFI FaxMaker. For more information, refer to Enabling XMLAPI in GFI FaxMaker (page 13).

» Create XMLAPIs using fields and parameter accepted by GFI FaxMaker. For more information, refer to Creating XMLAPIs (page 13).

3.1 How XMLAPI works

Screenshot 4: How XMLAPI works

Step	Description
1	**Generate XML API** Usually a 3rd party application is configured to automatically generate XML APIs. Ensure that the generated file complies to the parameters, fields and requirements of GFI FaxMaker. For more information, refer to Creating XMLAPIs (page 13).
2	**Store the API in XMLAPI pickup folder** In GFI FaxMaker, configure the folders from where to pick up APIs from. Store generated API in the appropriate folder. For more information, refer to Enabling XMLAPI in GFI FaxMaker (page 13).
3	**GFI FaxMaker retrieves API** Periodically, GFI FaxMaker polls configured folders for *.xml APIs.
4	**GFI FaxMaker processes and transmits fax/SMS** GFI FaxMaker processes the retrieved APIs and transmits the fax/SMS according to the fields and parameters specified in the API.

NOTE

Store attachments and/or body file as configured within the XML parameters of the API. Ensure that attachments and body files are fully stored in the specified location before storing the XMLAPIs in the pickup folder. If attachments or body files are not accessible by GFI FaxMaker during pickup, the fax fails.

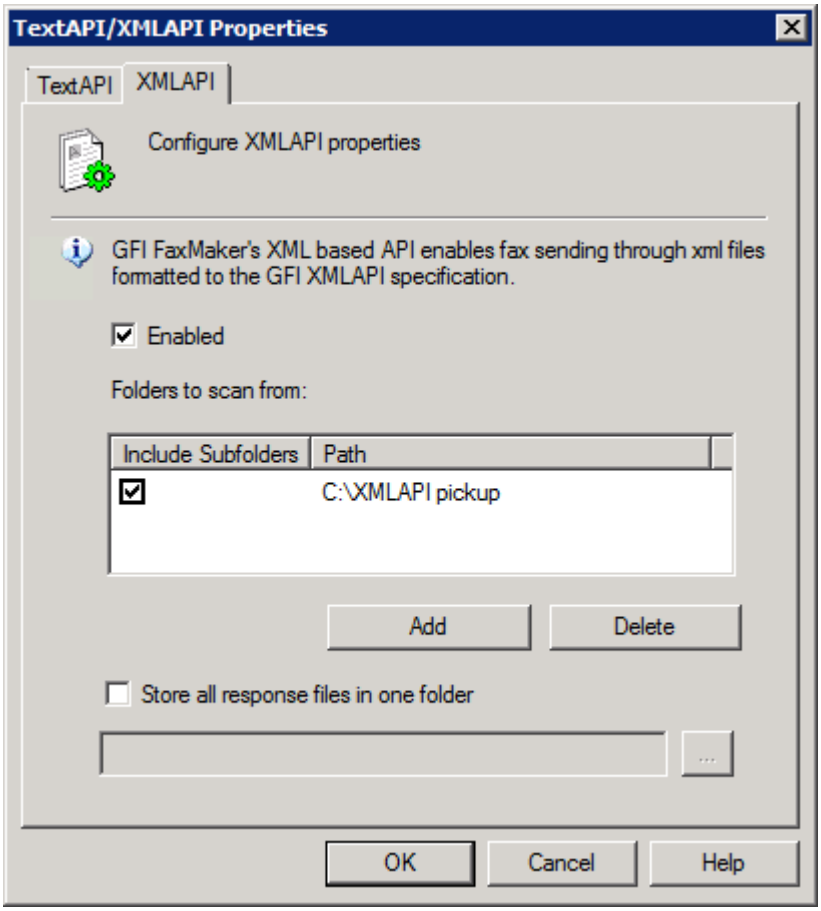
NOTE

After transmitting a fax using XMLAPI, GFI FaxMaker generates a response message (*.status file) in the XMLAPI folder which includes details about the fax transmission. This can be used by users, administrators or 3rd party applications to monitor fax transmission. For more information, refer to XMLAPI response files (page 20).

3.2 Enabling XMLAPI in GFI FaxMaker

To enable and configure XMLAPI:

1. From GFI FaxMaker Configuration, right-click **Advanced > Text-API/XMLAPI** node and select **Properties**.



Screenshot 5: Enabling XMLAPI

2. From **XMLAPI** tab, click **Enabled** to switch on XMLAPI.

3. Click **Add** to select the folders where XMLAPIs will be stored. To also check sub-folders for XMLAPIs, select **Include Subfolders**.

4. After processing an XMLAPI, GFI FaxMaker creates a response file (*.status extension). By default, GFI FaxMaker stores response files in the pickup folders. To store all response files in one location, select **Store all response files in one folder** and specify a location. For more information, refer to XMLAPI response files (page 20).

5. Click **OK**

Next step: Start creating XML APIs. For more information, refer to Creating XMLAPIs (page 13).

3.3 Creating XMLAPIs

When using XMLAPI, create an XML file (.xml format) that contains the parameters of the fax/SMS to transmit. All XMLAPI parameters should be within an XML container `<faxmakerdata>`. Data should then be contained in three separate containers as follows:

XMLAPI containers	Description
<fields>	In this container, specify fax transmission properties such as Subject, attachment and message file.
<sender>	Enter sender details in this container.
<recipients>	Specify recipient information in this container.

NOTE
An XML header is also required. This is usually:
`<?xml version="1.0" encoding="utf-8"?>`

For more information about XMLAPI fields and parameters, go to:

» Fax fields and parameters

» SMS fields and parameters

Sample XMLAPIs:

» Sample fax XMLAPI - http://go.gfi.com/?pageid=FM19help&lang=en#cshid=xmlapifax

» Sample SMS XMLAPI - http://go.gfi.com/?pageid=FM19help&lang=en#cshid=xmlapisms

3.4 Fax fields and parameters

List of supported fields and parameters in XMLAPI for sending faxes.

Message fields

Sender information

Fax recipient information

Refer to the sample fax XMLAPI for reference: http://go.gfi.com/?pageid=FM19help&lang=en#cshid=xmlapifax

3.4.1 Message fields

In the <fields> container, specify the following information:

Subject	**XML Path**	\faxmakerdata\fields\subject
	Type	String
	Disposition	Optional
	Details	Specify the subject of the fax.
	Example	<subject>This is a fax subject</subject>

Schedule	**XML Path**	\faxmakerdata\fields\schedule
	Type	String
	Disposition	Optional
	Details	Specify the date and time when to send the fax, in the following format: `YYYY-MM-DD-HH:mm:ss` Replace: » YYYY - year » MM - month » DD - day » hh - hours in 24-hour clock format » mm - minutes » ss - seconds
	Example	`<schedule>2021-10-10-08:15:00</schedule>`
Priority	**XML Path**	\faxmakerdata\fields\priority
	Type	String
	Disposition	Optional
	Details	Set message priority. Valid values are: » low » normal » high » veryhigh
	Example	`<priority>high</priority>`
Billing code	**XML Path**	\faxmakerdata\fields\billingcode
	Type	String
	Disposition	Optional. Required when Billing is enabled.
	Details	Sets the sender's billing code of the message.
	Example	`<billingcode>25</billingcode>`
Resolution	**XML Path**	\faxmakerdata\fields\resolution
	Type	String
	Disposition	Optional
	Details	Use this parameter to specify the resolution of the fax. Valid values are: » `Normal` (200 * 100 dpi) » `High` (200 * 200 dpi)
	Example	`<resolution>high</resolution>`
Coverpage	**XML Path**	\faxmakerdata\fields\coverpage
	Type	String
	Disposition	Optional
	Details	By default, fax is sent using the default coverpage configured in GFI FaxMaker. Use this parameter to send fax using a particular coverpage. Key in the coverpage to use, using one of the following: » Name of coverpage as configured in GFI FaxMaker. » Filename of a coverpage stored in *<GFI FaxMaker installation folder>*\Coverpage. For example, `generic.htm`.
	Example	`<coverpage>corporate</coverpage>`

Front covernote	**XML Path**	\faxmakerdata\fields\frontcovernote
	Type	String
	Disposition	Optional
	Details	By default, fax is sent using the default Front covernote configured in GFI FaxMaker. Use this parameter to send fax using a particular Front covernote. Key in the Front covernote to use, using one of the following: » Name of Front covernote as configured in GFI FaxMaker. » Filename of a Front covernote stored in *<GFI FaxMaker installation folder>*\Coverpage. For example, `MyCovernote.htm`.
	Example	`<frontcovernote>letterhead</frontcovernote>`
Fax line	**XML Path**	\faxmakerdata\fields\faxline
	Type	Number
	Disposition	Optional
	Details	Use this parameter to request the fax to be sent on a particular fax line. Specify the fax line number as configured in GFI FaxMaker Configuration. **NOTE:** If the line is busy or unavailable, the fax is sent on another line. To send the fax strictly from a particular line, use the following attribute: `demandline="true"`
	Example	`<faxline demandline="true">3</faxline>`
Header	**XML Path**	\faxmakerdata\fields\faxheader
	Type	String
	Disposition	Optional
	Details	Enter a custom personal header that is added to the fax header at the top of the fax.
	Example	`<faxheader>This is a fax header</faxheader>`
Unique ID	**XML Path**	\faxmakerdata\fields\uid
	Type	String [255 Chars. max.]
	Disposition	Optional
	Details	Specify an identifier that is used to mark the message. Uniqueness of this identifier is up to the creator of the message. Maximum number of characters is 255.
	Example	`<uid>3a5a90s7de3g20k9y4d5e7e6fse28se5e097</uid>`
TextAPI	**XML Path**	\faxmakerdata\fields\textapi
	Type	String
	Disposition	Optional
	Details	Used to specify a single line of TextAPI commands (including the '::' prefix). This field can be specified multiple times.
	Example	`<textapi>::billingcode=25,line=3</textapi>`

Message body file	**XML Path**	\faxmakerdata\fields\bodyfile
	Type	Full path or filename
	Disposition	Optional **NOTE:** Either an attachment or a fax message body file is required.
	Details	Uses the contents of the file as message body in the fax coverpage. Specify the body file in the following format: » **Full path:** Specify the full path (including the file name) of the body file. The file is automatically deleted on pickup. » **File name only:** Enter just the filename if the body file is stored in the coverpages folder: *<GFI FaxMaker installation folder>*\Coverpage. In this case, the body file is not deleted on pickup. **NOTE:** Ensure that the body file is fully stored in the appropriate location before storing the XMLAPI files in the pickup folder. If the body file is not accessible by GFI FaxMaker during pickup, the fax fails. Use the **type** attribute to specify the type of body file. Use one of the following attributes: » `type="text/html"` - Specify this attribute when message body text is in HTML format. » `type="text/rtf"` - Specify this attribute when message body text is in RTF format. Use RTF message body files with RTF coverpages only. » `type="text/plain"` - Specify this attribute when message body text is in plain text format. If no file type attribute is specified, message is entered as plain text.
	Example	`<bodyfile type="text/plain">c:\temp\bodyfile.txt</bodyfile>`
Attachment	**XML Path**	\faxmakerdata\fields\attachment
	Type	Full path or filename
	Disposition	Optional (maximum 60 attachments) **NOTE:** Either an attachment or a fax message body file is required.
	Details	Attaches the contents of the file specified to the fax. Specify the body file in the following format: » **Full path:** Specify the full path (including the file name) of the attachments. » **File name only:** Enter just the filename if attachments are stored in the same folder as the XMLAPI. **NOTE:** Ensure that attachment files are fully stored in the appropriate location before storing the XMLAPI files in the pickup folder. If attachments are not accessible by GFI FaxMaker during pickup, the fax fails. **NOTE:** Attachments are deleted after pickup,
	Example	`<attachment>c:\temp\document.doc</attachment>`

3.4.2 Sender fields

In the `<sender>` container, specify the sender information as follows:

First name	**XML Path**	\faxmakerdata\sender\firstname
	Type	String
	Disposition	Optional
	Details	Sets the first name of the sender.
	Example	`<firstname>John</firstname>`

Last name	**XML Path**	\faxmakerdata\sender\lastname
	Type	String
	Disposition	Optional
	Details	Sets the last name of the sender.
	Example	`<lastname>Smith</lastname>`
Company	**XML Path**	\faxmakerdata\sender\company
	Type	String
	Disposition	Optional
	Details	Sets the company name of the sender.
	Example	`<company>MyCompany Name</company>`
Department	**XML Path**	\faxmakerdata\sender\department
	Type	String
	Disposition	Optional
	Details	Sets the department of the sender.
	Example	`<department>Sales</department>`
Phone number	**XML Path**	\faxmakerdata\sender\voicenumber
	Type	String
	Disposition	Optional
	Details	Sets the voice (phone) number of the sender.
	Example	`<voicenumber>+1(800)1234 5678</voicenumber>`
Email address	**XML Path**	\faxmakerdata\sender\emailaddress
	Type	String
	Disposition	Required
	Details	Sets the email address of the sender.
	Example	`<emailaddress>jsmith@mydomain.com</emailaddress>`

3.4.3 Fax recipient fields

In the `<recipient>` container, specify the recipient information. For faxes, add a `<fax>` container and a `<recipient>` container.

For example:

```
<recipients>
   <fax>
      <recipient>
         <faxnumber>+1 800 1234 5678</faxnumber>
      </recipient>
   </fax>
```

```
</recipients>
```

> **NOTE**
> A fax can also be sent to multiple recipients. For more information, refer to Sending a fax to multiple recipients (page 20).

Insert other parameters and fields in the <recipient> container as follows:

First name	**XML Path**	\faxmakerdata\recipients\fax\recipient\firstname
	Type	String
	Disposition	Optional
	Details	Sets the first name of the recipient.
	Example	`<firstname>Joe</firstname>`
Last name	**XML Path**	\faxmakerdata\recipients\fax\recipient\lastname
	Type	String
	Disposition	Optional
	Details	Sets the last name of the recipient.
	Example	`<lastname>Bloggs</lastname>`
Company	**XML Path**	\faxmakerdata\recipients\fax\recipient\company
	Type	String
	Disposition	Optional
	Details	Sets the company name of the recipient.
	Example	`<company>FaxRecipient Company Ltd.</company>`
Department	**XML Path**	\faxmakerdata\recipients\fax\recipient\department
	Type	String
	Disposition	Optional
	Details	Sets the department of the recipient.
	Example	`<department>Marketing</department>`
Fax number	**XML Path**	\faxmakerdata\recipients\fax\recipient\faxnumber
	Type	String
	Disposition	Required
	Details	Sets the fax number of the recipient.
	Example	`<faxnumber>+1(800)8765 4321</faxnumber>`
Phone number	**XML Path**	\faxmakerdata\recipients\fax\recipient\voicenumber
	Type	String
	Disposition	Optional
	Details	Sets the telephone (voice) number of the recipient.
	Example	`<voicenumber>+1(800)8765 4444</voicenumber>`

Email address	**XML Path**	\faxmakerdata\recipients\fax\recipient\emailaddress
	Type	String
	Disposition	Optional
	Details	Sets the email address of the recipient.
	Example	`<emailaddress>John@faxrecipientdomain.com</emailaddress>`

Sending a fax to multiple recipients

A fax can be sent to multiple recipients. To do this, include multiple `<recipient>...</recipient>` containers. For example:

```
<recipients>
   <fax>
      <recipient>
         <firstname>Bob</firstname>
         <faxnumber>+1 800 1111 1111</faxnumber>
      </recipient>
      <recipient>
         <firstname>John</firstname>
         <faxnumber>+1 800 2222 2222</faxnumber>
      </recipient>
   </fax>
</recipients>
```

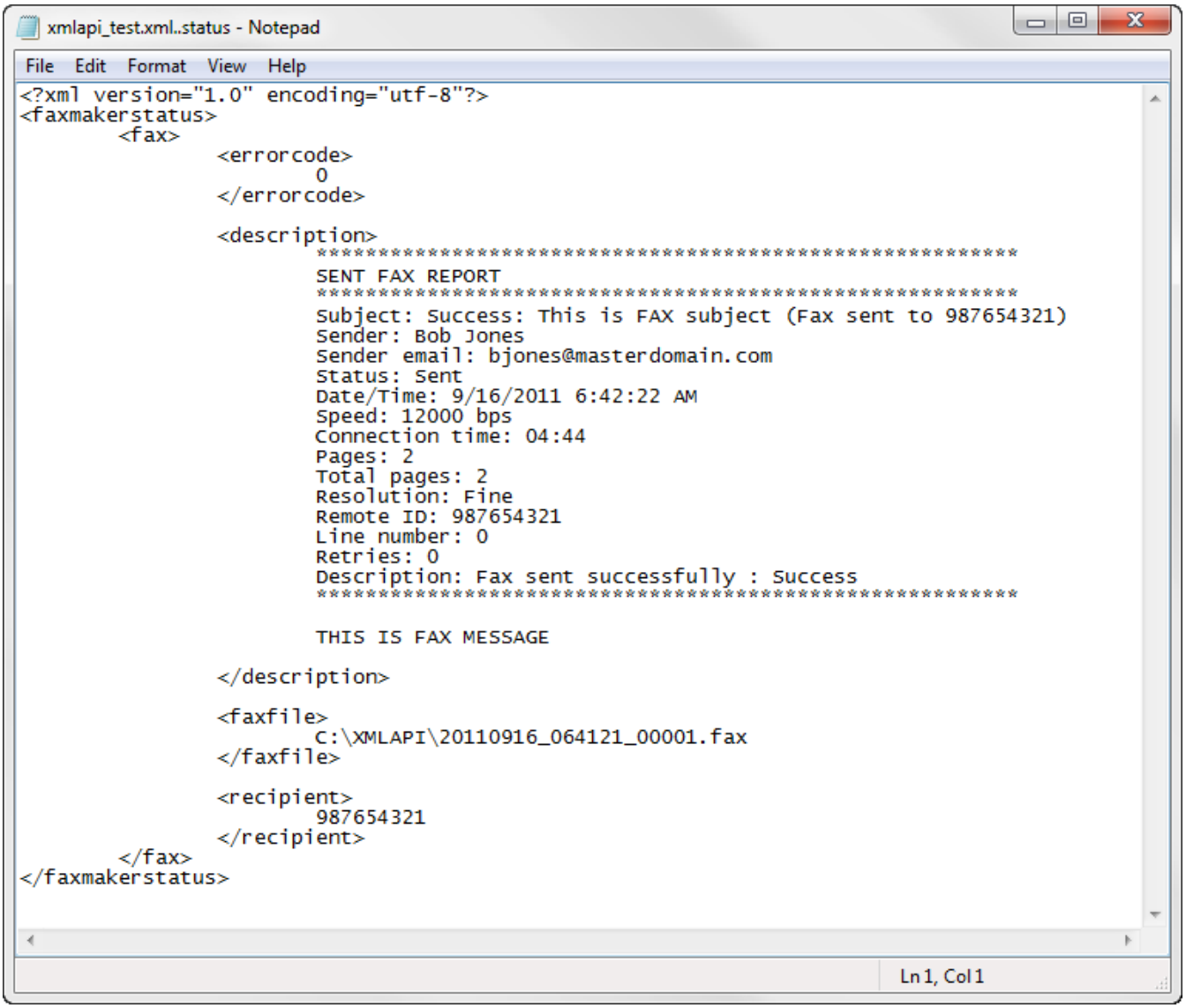
3.4.4 XMLAPI response files

After processing an XMLAPI, GFI FaxMaker creates a response file which can be used to review fax transmission. 3rd party tools can be configured to poll this folder and automatically retrieve the status of transmitted faxes.

By default, GFI FaxMaker stores response files in the same folder where the XMLAPI was originally picked up. Alternatively, GFI FaxMaker can be configured to store all response files in a particular folder. For more information, refer to Enabling XMLAPI in GFI FaxMaker (page 13).

The name of the response file matches the name of the xmlapi, with a ***.status** extension. For example, the response file of XMLAPI **fax123.xml**, is named **fax123.xml.status**. If a response file with the same name already exists, GFI FaxMaker appends the response message to the file.

> **NOTE**
> SMS transmission does not generate a response file.



Screenshot 6: Sample XMLAPI response message

The response message contains the following XML containers:

Container	Description
<fax>...</fax>	Container for each fax transmission. If one XML API file contains multiple recipients, a <fax> container is created for each recipient, containing all details for that particular transmission.
<errorcode>...</errorcode>	Returns the error code for fax transmission. A successful transmission returns a **0** error code.
<description>...</description>	Contains transmission report for successful or failed transmissions. This includes important parameters about the fax.
<uid>...</uid>	If a unique ID (uid) was specified in the XMLAPI, this container is produced.
<faxfile>...</faxfile>	Contains full path to image file of the fax.
<recipient>...</recipient>	Displays the fax number to which the fax has been sent.

3.5 SMS fields and parameters

List of supported fields and parameters in XMLAPI for sending SMS.

Message fields

Sender information

SMS recipient information

Refer to the sample SMS XMLAPI for reference: http://go.gfi.com/?pageid=FM19help&lang=en#cshid=xmlapisms

3.5.1 Message fields

In the `<fields>` container, specify the following information:

SMS message	**XML Path**	\faxmakerdata\fields\bodyfile
	Type	Full path or filename
	Disposition	Required
	Details	Uses the contents of the file as the SMS message. Key in the full path of the file. Specify the file type: » `type="text/html"` - message body text is in HTML format. » `type="text/rtf"` - message body text is in RTF format. » `type="text/plain"` - message body text is in plain text format. If no file type attribute is specified, message is processed as plain text.
	Example	`<bodyfile type="text/plain">c:\temp\bodyfile.txt</bodyfile>`
SMS line	**XML Path**	\faxmakerdata\fields\gsmline
	Type	Number (COM Port) or String (name configured for GSM line)
	Disposition	Optional
	Details	Request an SMS to be sent on a particular GSM line. Specify the name or the COM Port number as configured in GFI FaxMaker. Use the `demandline` parameter to specify how to send the SMS if the specified line is busy or does not exist. Use `demandline="true"` to send the SMS strictly from the specified GSM line. Use `demandline="false"` to send the SMS via the next available GSM line if the specified line is busy or does not exist.

	Example	`<gsmline demandline="true">3</gsmline>`
		`or`
		`<gsmline demandline="false">MyGSMline</gsmline>`

3.5.2 Sender fields

In the `<sender>` container, specify the sender information as follows:

First name	**XML Path**	\faxmakerdata\sender\firstname
	Type	String
	Disposition	Optional
	Details	Sets the first name of the sender.
	Example	`<firstname>John</firstname>`
Last name	**XML Path**	\faxmakerdata\sender\lastname
	Type	String
	Disposition	Optional
	Details	Sets the last name of the sender.
	Example	`<lastname>Smith</lastname>`
Company	**XML Path**	\faxmakerdata\sender\company
	Type	String
	Disposition	Optional
	Details	Sets the company name of the sender.
	Example	`<company>MyCompany Name</company>`
Department	**XML Path**	\faxmakerdata\sender\department
	Type	String
	Disposition	Optional
	Details	Sets the department of the sender.
	Example	`<department>Sales</department>`
Phone number	**XML Path**	\faxmakerdata\sender\voicenumber
	Type	String
	Disposition	Optional
	Details	Sets the voice (phone) number of the sender.
	Example	`<voicenumber>+1(800)1234 5678</voicenumber>`
Email address	**XML Path**	\faxmakerdata\sender\emailaddress
	Type	String
	Disposition	Required
	Details	Sets the email address of the sender.
	Example	`<emailaddress>jsmith@mydomain.com</emailaddress>`

3.5.3 SMS recipient fields

In the `<recipient>` container, specify the recipient information. For SMS, add a `<sms>` container and a `<recipient>` container.

For example:

```
<recipients>
   <sms>
      <recipient>
         <smsnumber>+1 800 1234 5678</smsnumber>
      </recipient>
   </sms>
</recipients>
```

NOTE
An SMS can also be sent to multiple recipients. For more information, refer to Sending an SMS to multiple recipients (page 25).

Insert other parameters and fields in the <recipient> container as follows:

First name	**XML Path**	\faxmakerdata\recipients\sms\recipient\firstname
	Type	String
	Disposition	Optional
	Details	Sets the first name of the recipient.
	Example	`<firstname>Joe</firstname>`
Last name	**XML Path**	\faxmakerdata\recipients\sms\recipient\lastname
	Type	String
	Disposition	Optional
	Details	Sets the last name of the recipient.
	Example	`<lastname>Bloggs</lastname>`
Company	**XML Path**	\faxmakerdata\recipients\sms\recipient\company
	Type	String
	Disposition	Optional
	Details	Sets the company name of the recipient.
	Example	`<company>SMSRecipient Company Ltd.</company>`
Department	**XML Path**	\faxmakerdata\recipients\sms\recipient\department
	Type	String
	Disposition	Optional
	Details	Sets the department of the recipient.
	Example	`<department>Marketing</department>`

SMS number	**XML Path**	\faxmakerdata\recipients\sms\recipient\smsnumber
	Type	String
	Disposition	Required
	Details	Sets the sms number of the recipient.
	Example	`<smsnumber>+1(800)8765 4321</smsnumber>`
Voice number	**XML Path**	\faxmakerdata\recipients\sms\recipient\voicenumber
	Type	String
	Disposition	Optional
	Details	Sets the telephone (voice) number of the recipient.
	Example	`<voicenumber>+1(800)8765 4444</voicenumber>`
Email address	**XML Path**	\faxmakerdata\recipients\sms\recipient\emailaddress
	Type	String
	Disposition	Optional
	Details	Sets the email address of the recipient.
	Example	`<emailaddress>John@smsrecipientdomain.com</emailaddress>`

Sending an SMS to multiple recipients

An SMS can be sent to multiple recipients. To do this, include multiple `<recipient>...</recipient>` containers. For example:

```
<recipients>
   <sms>
      <recipient>
         <firstname>John</firstname>
         <smsnumber>+1 800 1111 1111</smsnumber>
      </recipient>
      <recipient>
         <firstname>Bob</firstname>
         <smsnumber>+1 800 2222 2222</smsnumber>
      </recipient>
   </sms>
</recipients>
```

4 Web Service API

The GFI FaxMaker Web Service API enables you to develop your own applications that can connect and send/receive faxes via GFI FaxMaker.

> **NOTE**
> This topic is aimed at users that are familiar with Visual Studio, XML Web Services and SOAP technologies.

4.1 Requirements and important notes

» To create the API, use a development environment that supports XML Web Services and SOAP.

» GFI FaxMaker Web Services API uses port 8555 for HTTP communication between the application and GFI FaxMaker. This port can be opened automatically when running the GFI FaxMaker Configuration Wizard. For more information refer to http://go.gfi.com/?pageid=FM19help&lang=en#cshid=FirewallPorts

» Access the GFI FaxMaker Web Service through XML Web Services over HTTP port 8555, using the URL: `http://<GFI FaxMaker Server>:8555/faxmaker/wsapi`
Replace *<GFI FaxMaker Server>* with the name or IP address of the GFI FaxMaker server. For example `http://myfaxserver.mydomain.local:8555/faxmaker/wsapi`.

» Alternatively you may use a secure connection over HTTPS. For more information refer to http://go.gfi.com/?pageid=FM19help&lang=en#cshid=WSAPIhttps

» From the **Licensed Users** node register users that will use the Web Service API. Use these users' credentials in API calls.

» Received faxes that are routed to users registered for Web Service API are stored in the Web Service API queue and not forwarded to the user via email. Configure routing rules to route faxes that are to be picked up by the Web Service API to the registered users.

» GFI FaxMaker retains received faxes in the Web Service API queue for 30 days. Expired faxes cannot be fetched by the Web Service API. For more information, refer to Managing the Web Service API queue (page 35).

4.2 Getting started

In Visual Studio, create a C# Windows or Console Application and from Solution Explorer, add a web or service reference (depending on the Visual Studio version).

When prompted to enter the service URL, key in the GFI FaxMaker Web Service URL WSDL link. This is composed of the GFI FaxMaker Web Service URL as defined in the Requirements section above and add `?singleWsdl`. For example, `http://myfaxserver.mydomain.com:8555/faxmaker/wsapi?singleWsdl`

When prompted to enter the namespace or a web reference name, key in `FMWSAPI`.

After creating the application, open **app.config**. Find the address `http://127.0.0.1:8555/faxmaker/wsapi` and change it with the value of your GFI FaxMaker Web Service URL. For example, `http://myfaxserver.mydomain.com:8555/faxmaker/wsapi`

Save changes and proceed to use the following namespace classes as typical C# classes.

4.3 API Functions

4.3.1 SendFax

Use this function to send a fax. The function returns an array of `FaxJobIDs`. For every recipient specified, a `FaxJobID` is created. This ID gives the ability to monitor the status of that fax sent to that particular recipient.

```
FMResult SendFax
(
   FMUser user,
   UserDetails senderdetails,
```

```
    List<UserDetails> recipients,
    MessageDetails messagedetails,
    out FaxJobID[] faxjobids
);
```

Parameters:

Parameter	Description
user	The web API user credentials.
senderdetails	The details of the user.
recipients[]	List of recipients of the fax.
messagedetails	The fax to be send.

Return Values:

Parameter	Description
FMResult	Return the result of the function.
Faxjobids	An array of faxjobid. For every recipient there is a job id. This job id can be later used to get the status of a particular fax.

4.3.2 CheckForUnreadFaxes

Use this function to get the list of faxes in the queue. This queue contains all those faxes which were not previously downloaded.

Every element in this list contains only the details of that fax and not the actual fax image. To download the fax use `GetNextFax()`.

```
FMResult CheckForUnreadFaxes
(
    FMUser user,
    out ReceivedFaxDetails[] receivedfaxes
);
```

Parameters:

Parameter	Description
user	The web API user credentials.

Return Values:

Parameter	Description
FMResult	Return the result of the function.
receivedfaxes	An array of ReceivedFaxDetails.

4.3.3 GetSendingFaxStatusUpdates

Use this function to obtain the status of a fax which was previously submitted using the SendFax API.

```
FMResult GetSendingFaxStatusUpdates
(
   FMUser user,
   long FaxJobID,
   out FaxSendingStatus FaxStatus
);
```

Parameters:

Parameter	Description
user	The web API user credentials.
FaxJobID	The unique identifier assigned to each recipient of an outbound fax.

Return Values:

Parameter	Description
FMResult	Return the result of the function.
FaxStatus	Returns the status of the fax.

4.3.4 GetNextFax

Use this method to get the oldest fax from the queue. This queue contains all those faxes which were not previously downloaded.

When getting the oldest fax, it is not automatically deleted from queue. Call `DeleteFax()` to delete the oldest fax, enabling you to get the next fax. If the oldest fax is not deleted you will receive the same fax again.

```
FMResult GetNextFax
(
   FMUser user,
   out ReceivedFaxDetails faxdetails,
   out FileData fax
);
```

Parameters:

Parameter	Description
user	The web API user credentials.

Return Values:

Parameter	Description
FMResult	Return the result of the function.
faxdetails	A ReceivedFaxDetails which hold the details of the fax
fax	The fax image.

4.3.5 GetNextFaxByCriteria

Use this method to get the oldest fax from the queue, according to the criteria specified. This queue contains all those faxes which were not previously downloaded.

When getting the oldest fax, it is not automatically deleted from queue. Call `DeleteFax()` to delete the oldest fax, enabling you to get the next fax. If the oldest fax is not deleted you will receive the same fax again.

```
FMResult GetNextFaxByCriteria
(
   FMUser user,
   SearchCriteria criteria,
   stringcriteriavalue,
   out ReceivedFaxDetails faxdetails,
   out FileData fax
);
```

Parameters:

Parameter	Description
user	The web API user credentials.
Criteria	DTMF or Remote ID
Criteriavalue	The value of the criteria

Return Values:

Parameter	Description
FMResult	Return the result of the function.
faxdetails	A ReceivedFaxDetails which hold the details of the fax
Fax	A FileData contains the actual fax image data.

4.3.6 DeleteFax

Use this function to delete a particular fax.

```
FMResult DeleteFax
(
   FMUser user,
```

```
    int FaxID
);
```

Parameters:

Parameter	Description
`user`	The web API user credentials.
`FaxID`	The ID of the fax to delete. This value can be retrieved from the output of `CheckForUnreadFaxes()`, `GetNextFax()` or `GetNextFaxByCriteria()`.

Return Values:

Parameter	Description
`FMResult`	Return the result of the function.

4.4 Classes

4.4.1 UserDetails

This class contains the details of a recipient/sender.

All fields are optional, except for `Faxnumber` property when used as a recipient.

Data members	Type	Description
`Firstname`	String	The first name.
`Lastname`	String	The last name.
`Company`	String	Company name.
`Department`	String	Department name.
`Faxnumber`	String	The fax number.
`Email`	String	The email address.

4.4.2 MessageDetails

An object of this class holds the details of a new fax.

All fields are optional, except for the `BillingCode` when Billing is enabled.

Data members	Type	Description
`Subject`	String	The fax subject.
`Schedule`	DateTime	A specific date & time when to send the fax.
`Priority`	FaxPriority	Set the fax's priority.
`BillingCode`	String	Specifies the billing code. Required when Billing is enabled.

Data members	Type	Description
`Resolution`	FaxResolution	The resolution to use.
`Coverpage`	String	By default, a fax is sent using the default coverpage configured in GFI FaxMaker. Use this parameter to send fax using a particular coverpage.
`FrontCoverNote`	String	By default, fax is sent using the default Front covernote configured in GFI FaxMaker. Use this parameter to send fax using a particular Front covernote.
`FaxLine`	Integer	Use this parameter to request the fax to be sent on a particular fax line. Specify the fax line number as configured in GFI FaxMaker Configuration.
`UseFaxLine`	FaxLineType	Set to FaxLine or demand line if required. Otherwise, set to none. The line to be used has to be specified with the FaxLine property.
`header`	String	Enter a custom personal header that is added to the fax header at the top of the fax.
`TextAPI`	String	Specifies a single line of TextAPI command. For a list of TextAPI commands, refer to TextAPI commands.
`MessageBodyFile.`	FileData	An instance of FileData, holding a text file containing the message body of the fax.
`Attachments`	List< FileData>	Array of attachments.

4.4.3 FMUser

A Web API user.

Data members	Type	Description
`Email`	String	The user's email address as configured in the GFI FaxMaker Configuration Licensed users node. The user must be a registered Web API user.
`Password`	String	In Active Directory, this is the configured user's password. In SMTP environments the password is an auto generated GUID available from the GFI FaxMaker Configuration Licensed users node for registered Web API users.

4.4.4 FaxJobID

GFI FaxMaker assigns a unique identifier to each recipient of an outbound fax. Use this identifier to monitor the status of a fax.

Data members	Type	Description
`Recipient`	String	The recipient's fax number.
`ID`	String	The unique identifier assigned by GFI FaxMaker.

4.4.5 ReceivedFaxDetails

An object of this class is received when `GetNextFax()` is called.

Data members	Type	Description
`FaxUID`	String	Fax unique identifier. Use this identifier when deleting a fax.

Data members	Type	Description
WhenReceived	DateTime	Date and time when fax was received.
NosPages	String	Number of pages received.
Resolution	FaxResolution	The received fax resolution.
RemoteID	String	The sender's fax machine name.
DTMF_DID	String	The DTMF/DID extension number of the receiving line.

4.4.6 FileData

An object of such class holds the actual data of a file.

Data members	Type	Description
Filename	String	The file name of the file; mainly used to determine the file format.
Data	Byte[]	The file data in a byte array.

4.4.7 FaxSendingStatus

An object of this class is returned when `GetSendingFaxStatusUpdates()` is called.

Data members	Type	Description
FaxUID	String	Fax unique identifier.
RemoteID	String	The sender's fax machine name.
CurrentTransmissionPage	Integer	The page that is currently being transmitted. After a fax is sent, this returns the total number of pages transmitted.
TotalPages	Integer	The total number of pages that make up the fax.
Status	SendingStatus	The current status of the fax.

4.5 Enums

4.5.1 FaxPriority

Value	Description
None	No priority specified.
Low	Low priority.
Normal	Normal priority.
High	High priority.
VeryHigh	Very high priority.

4.5.2 FaxLineType

Value	Description
None	No demand fax line required.
Faxline	Reserved.
Demand	Send the fax strictly from a particular line (FaxLine attribute)

4.5.3 FaxResolution

Value	Description
NonSet	No custom fax resolution specified.
High	High resolution.
Normal	Normal fax resolution.

4.5.4 SearchCriteria

Value	Description
RemoteID	Search faxes by caller ID or fax machine name.
DTMF	Search faxes by the extension number of the receiving line.

4.5.5 SendingStatus

Value	Description
Preparing	Processing and converting the content to fax.
ConvertingAttachment	Converting the attachment to fax format.
Pending	The fax is in queue, ready to be faxed.
Sending	Fax in transmission.
Sent	Fax was sent.
Failed	Transmission failed.
Cancelled	The fax job was cancelled.

4.5.6 FMResult

Value	Description
Success	Function completed successfully.
Failure	Function failed.
UserNotValid	The user details are not valid.
DBError	Error writing to or reading from database.
FaxImageNotFound	The required fax image was not found.

Value	Description
`NoSender`	No or invalid sender specified.
`NoRecipients`	No or invalid recipient specified.
`InvalidFaxDescription`	One or more required fields were not supplied.

To help you get started with developing custom applications for GFI FaxMaker Web Service API, refer to the samples provided.

» Java sample: http://go.gfi.com/?pageid=FM19help&lang=en#cshid=javaWsapi

» C# sample: http://go.gfi.com/?pageid=FM19help&lang=en#cshid=netWsapi

4.6 Managing the Web Service API queue

When a fax is routed to a registered Web Service API user, GFI FaxMaker stores the fax in the Web Service API queue.

Use the various functions documented in Web Service API to get faxes from the queue. When getting a fax, it is not automatically deleted from queue. Call `DeleteFax()` to delete it.

4.6.1 Expired faxes in queue

Received faxes are stored in the Web Service API queue for 30 days. If these exceed 30 days, faxes are automatically moved from the queue and cannot be fetched by the Web Service API.

By default, expired faxes are moved to the following folder:

<GFI FaxMaker installation folder>`/WSAPI/Retention/`

For each expired fax moved to this folder, GFI FaxMaker stores the fax image in .fax format (can be opened in an image viewer) and an XML file containing the fax meta data.

To change the default folder:

1. Stop all GFI FaxMaker services.

2. Navigate to the GFI FaxMaker installation folder.

3. Open file **ML.Svc.Attendant.exe.config** in a text editor.

4. Find key **RetentionFolder** and change its value to the new path.

5. Start the services that were stopped in step 1.

> **NOTE**
> Use caution when modifying the software's installation files. Misconfiguration can cause problems in the product's functionality.

5 Troubleshooting and support

This topic explains how to resolve any issues encountered during installation of GFI FaxMaker. The main sources of information available to solve these issues are:

- This manual - most issues can be solved through the information in this help system.
- GFI Knowledge Base articles
- Web forum
- Contacting GFI Technical Support

5.1 Other troubleshooting resources

5.1.1 Knowledge Base

GFI maintains a comprehensive Knowledge Base repository, which includes answers to the most common installation problems. In case that the information in this manual does not solve your installation problems, next refer to the Knowledge Base. The Knowledge Base always has the most up-to-date listing of technical support questions and patches. Access the Knowledge Base by visiting: https://www.gfi.com/support/products/gfi-faxmaker/.

5.1.2 Web Forum

User to user technical support is available via the GFI web forum. Access the web forum by visiting: http://forums.gfi.com/.

5.1.3 Request technical support

If none of the resources listed above enable you to solve your issues, contact the GFI Technical Support team by filling in an online support request form or by phone.

- **Online**: Fill out the support request form and follow the instructions on this page closely to submit your support request on: https://www.gfi.com/support/technical-support-form
- **Phone**: To obtain the correct technical support phone number for your region visit: https://www.gfi.com/contact-us

> **NOTE**
> Before contacting Technical Support, have your Customer ID available. Your Customer ID is the online account number that is assigned to you when first registering your license keys in the GFI Customer Area at: http://customers.gfi.com.

We will answer your query within 24 hours or less, depending on your time zone.