Adobe FrameMaker

FDK Installation Guide

Adobe

September 2022

Contents

Enter title here

About the FDK

Easily author and publish technical content for mobile, web, desktop, and print.

Best-in-class authoring and publishing software

The Adobe FrameMaker Developer Kit (FDK) is a set of C libraries and header files that allow you to write Application Programming Interface (API) clients or plug-ins using the C or C++ programming languages. Using the FDK, an API client can perform interactive user tasks and more. It gives programmers direct access to all the objects in a FrameMaker session or document. Gain access to common objects like graphics, text, markers, paragraphs, formats, and more.

The guides, documentation, and sample code in the FDK show you how to write plug-ins using C or C++ and create plug-ins to automate complex tasks.

Download the FDK today and unleash the powerful capabilities of your favorite industry-standard tool for authoring and publishing multilingual technical content.

- **Automate complex workflows**

  Easily automate any complex workflow in your authoring lifecycle. Use ExtendScript - an extended implementation of JavaScript that provides a scripting interface for FrameMaker and many other Adobe applications. For more information, refer to ExtendScript API Reference.

- **Create your own structured applications**

  Control the way content is authored by creating structured applications. Use structured templates, XML or SGML applications to create an authoring ecosystem for your authors. For more information, refer to Structured Application Development Guide.

- **Connect with external word processing applications**

  Use FrameMaker's Maker Interchange Format (MIF) to convert any FrameMaker document into your word processing or publishing program's format. For more information, refer to MIF Reference Guide.

- **Leverage the power of FDK and FDE**

  Use the FDK to create powerful clients or plug-ins to meet your authoring needs. Along with the FDK, you also get FrameMaker Development Environment (FDE), which provides platform-independent alternatives to platform-specific I/O, string, and memory allocation schemes. For more information, refer to FDK Programmer's Guide.

- **Work with customized DITA**

DITA Specialization allows you to reuse elements from higher-level design and create custom information models to meet your business requirements. Also, customize document shells by removing domains or individual elements. For more information, refer to DITA 1.3 Customization.

- **Completely control authoring and publishing**

  The powerful and configurable (.ini) files give you complete control on the way you author or publish content. From the most granular settings, like setting the default zoom level for your document, to controlling the elements you want in your published output, you have complete control. For more information, refer to INI Reference.

- **Connect to any CMS**

  Integrate easily with the most popular Content Management Systems (CMS) like AEM, DITA Exchange, Documentum, and SharePoint. Or, create your own CMS connector using the FDK.

Enter title here

Download and install FrameMaker Developer Kit

The FrameMaker Developer Kit™ (FDK) provides tools for developers to enhance the functionality of FrameMaker. If you are using the latest version (FDK 2022), then you get a downloadable ZIP of the FDK.

This chapter provides you with the details of downloading and installing the FDK, and the files and folders created in the process.

Install through the FDK ZIP

To download and use the FDK 2022, perform the following steps:

1) Download the FDK ZIP from Adobe.io.
2) Save the ZIP file on your system and give it an appropriate name, such as `FDK2022`.
3) Unzip the contents of the ZIP file.

Once you unzip the contents of the ZIP file, you are ready to use the FDK. There's no explicit installation required to use the ZIP version of the FDK. For details of the files and folders created after unzipping, see .

Contents of the downloaded file

As you unzip or install the FDK ZIP, the following files and folders are created on your system:

- `Application` (folder)
- `SupportFiles` (folder)
- `readme.txt (file)`

The `SupportFiles` folder contains the `license.html` file (within `\Legal\en_US` folder), which contains the *Software License Agreement* for using the FDK.

The `readme.txt` file contains the FDK details like the FDK version, minimum supported FrameMaker version, and the required Visual Studio version.

The `Application` folder contains the FDK header files, libraries, and sample code. The following table lists the contents of this folder:

Folder	What it contains
`include`	Public header files that you include in FDK clients.

Folder	What it contains
lib	FDK libraries that you link with your client.
doc	A list of a list of sample clients and online documentation for the FDK in Adobe® Acrobat® PDF.
samples	Source code and project files for sample FDK clients and a dialog resource template.

include

The include folder contains FDK header files. The files are listed in the following table.

Folder	What it contains
f_local.h	Provides a location for declarations for your platform-specific functions.
f_stdio.h	Provides declarations for platform-independent C library functions.
f_types.h	Defines Frame Development Environment (FDE) fundamental data types. It isincluded in fapi.hand fdetypes.h
fapi.h	Provides definitions and function declarations for the FDK. You must include itin all FDK clients.
fapidefs.h	Defines constants you can use to specify objects, properties, and some functionarguments. It is included by fapi.h
fassert.h	Provides declarations for FDE assert functions.
fchannel.h	Provides declarations for FDE channel functions.
fcharmap.h	Provides declarations for FDE character functions.
fcodes.h	Provides declarations for function codes (f-codes).
fdetypes.h	Provides declarations for FDE data types. You must include it in all FDK clientsthat use the FDE.
fdk_env.h	Provides top-level header file for individual platforms.
fencode.h	Provides declarations for API and FDE font encoding functions.
fhash.h	Provides declarations for FDE hash functions.
fioutils.h	Provides declarations for FDE I/O utility functions.
fitstub.h	Provides declarations for filter functions.

Folder	What it contains
fm_base.h	Defines types and data structures for the Structure Import/Export API. It isincluded by fm_struct.h
fm_coma.h	Only present for backward compatibility. Use fcodes.h
fm_psr.h	Defines types and data structures for the Structure Import/Export API. It isincluded by fm_struct.h
fm_rdr.h	Defines types and data structures for the Structure Import/Export API. It isincluded by fm_struct.h
fm_sgml.h	Retained for backward compatibility—use fm_struct.hinstead.
fm_struct.h	Provides declarations for Structure Import/Export API functions. You mustinclude it in all Structure Import/Export API clients.
fm_wtr.h	Defines types and data structures for the Structure Import/Export API. It isincluded by fm_struct.h
fmemory.h	Provides declarations for FDE memory functions.
fmetrics.h	Provides declarations for FDE metric functions.
fmifdata.h	Provides declarations for FDE Maker Interchange Format (MIF) functions.
fmifmacr.h	Provides macros for writing MIF statements.
fmifname.h	Provides definitions for MIF statements.
fmifstmt.h	Provides declarations for FDE MIF statement functions.
fmifstrt.h	Provides MIF data structures.
fmiftype.h	Provides basic data structures used by MIF data structures.
fpath.h	Provides definitions used by filepath functions.
fprogs.h	Provides declarations for FDE progress functions.
fstdio.h	Provides declarations for FDE I/O functions.
fstrings.h	Provides declarations for FDE string functions.
fstrlist.h	Provides declarations for FDE string list functions.
fstrres.h	Provides internally used string resource functions. Do not include with yourFDK clients.
futils.h	Provides declarations for FDE utility functions.

lib

The lib folder contains the library files listed in the following table.

Folder	What it contains
api.lib	The API library. To use any API functions, you must link this library with yourclient.
fdk.lib	The FDE library. To use any FDE functions, you must link this library withyour client.
fmdbms32.lib	FDK heap management library. Link all FDK clients with this library.
cssparserinterface.lib	Provides CSS parsing and conversion to the corresponding Element Definition Document (EDD).
fmstruct.res	Provides SGML/XML resources. You must link all Structure Import/ExportAPI clients with it.
struct.lib	Provides Structure Import/Export API functions. You must link all StructureImport/Export API clients with it.

doc

The doc folder contains the FDK documentation for all platforms in Adobe Acrobat PDF.

PDF File	Description
fdkinstallguide.pdf	The FDK Installation Guide.
fdkprogrammerguide.pdf	The *FDK Programmer’s Guide*.
fdkreference.pdf	The *FDK Programmer’s Reference.*
samplelist.pdf	A list of the code samples shipped with the FDK, including brief descriptions of each one. Source code and project files for sample FDK clients and a dialog resource template.
structapi.pdf	The *Structure Import/ Export API Programmer’s Guide.*

Samples

The samples folder contains the code for sample clients and a sample dialog resource file. The samples\winsamp folder contains the code for a sample client that is specific to the Windows platform.

For a list of the samples that are included with the FDK, and a brief description of each one, see the online document `samplelist.pdf`.This file is included with the FDK in the doc folder.

The other folders in the `samples` folder (with the exception of `dre`) include one or more source (`.c`) files, appropriate header (`.h`) files and appropriate workspace, solution, and project file. All the sample code in those folders is platform independent. With an appropriate makefile, you can compile it on any of the platforms the FDK supports. For information about a client, see the comments at the beginning of the client's source (`.c`) file.

IMPORTANT: *Permission to use, reproduce, modify, and distribute the Sample Clients is for the sole purpose of integrating your software applications with Adobe Systems Incorporated ("Adobe") software ("Sample Clients" are defined as those files located in the `fdk\samples` folder). Such permission is hereby granted without fee, provided that*

(i) you distribute the Sample Clients only as part of your software application;

(ii) the copyright notice appears in and on all copies of your software application.

Enter title here

Legal notices

For legal notices, visit the Legal Notices page.