# OpenText File Content Extraction

Identify, extract and transform file contents

## Benefits

- Reduce time to market, engineering risk, internal development costs, and ongoing maintenance.
- Free up developers from spending time to stay current with the ever-changing landscape of file formats—1,950 currently supported.
- Reduce risk of misprocessing crucial information.
- Stop spending valuable CPU time on irrelevant files.
- Don't miss a thing, with the richest text extraction technology available.
- Extend visibility into all document sources with OCR.
- Lower latency by pipelining data at sub-document granularity.
- Improve usability by providing high-fidelity HTML renderings.
- Increase stability with KeyView's secure out-of-process capabilities.
- Support your customers' journey into emerging secure data sharing trends with support for rights management functionality, such as Microsoft Purview Information Protection.

Empower your customers to get more out of their data by providing accurate file format identification, content decryption, text extraction, subfile processing, non-native rendering, and structured export, with support for 1,950 formats across all major client and server-side platforms.

## Product Highlights

OpenText™ File Content Extraction is a mature and professionally maintained OEM-embeddable SDK for file format identification, content extraction and file transformation used by leading edge software developers and service providers to add competitive differentiation, and to significantly reduce the business risks associated with managing large volumes of human-originated information.

### Faster Time-to-Value

OpenText File Content Extraction is a low-risk way to incorporate deep content visibility to your service or application—quickly, reliably, and without the need for ongoing development.

We provide a ready-to-go SDK, complete with sample code to accelerate your product's time-to-market, freeing your engineering organization to spend their time on your business's core value proposition.

*We keep up to date with the busy world of new and evolving formats so you don't have to.*

### Broader Coverage

Consider the industries and regions into which you sell.

Established corporations need support for a wide variety of historic and modern office applications to ensure data coverage. Startups and newer companies often rely on newer cloud-native formats. If you target governments, especially in Europe or South America, good support for open formats can be a deal maker. Some industries have a particular class of file formats which are disproportionately important but not generally well supported elsewhere, such as the compliance market's requirement for historic office files.

Geographic diversity plays an important role when it comes to format support. Even in "easy" areas such as office applications, different regions have different de facto applications and file formats. For example, the Hangul office suite—with its own evolving set of formats—is widespread in South Korea due to its good support for the Korean language. In China, the domestic WPS Office suite (previously Kingsoft) competes with Microsoft Office.

Breadth of coverage applies to language support too: PDF documents containing right-to-left languages or languages with offset characters, such as vowels in Arabic and Hebrew, can prove difficult to extract accurate text from.

How do you plan to grow your business? Where will you target next? Is your solution prepared for the unique set of formats you will encounter in that market?

*With continuous development for more than 25 years, OpenText File Content Extraction has an extensive catalogue of supported formats, kept up to date by our professional team to address the relentlessly increasing number of relevant document formats—everything from legacy formats which current software cannot read, to formats from new applications and software updates, as well as nascent cloud-specific formats.*

## Deeper Coverage

Accurate file type identification is crucial for determining the correct downstream processing. File formats vary massively in terms of the amount and accessibility of usable content. It is important to correctly identify even file types which cannot contain useful content, to make an accurate risk assessment for use cases such as DLP and eDiscovery.

Metadata is extracted even when there is no useful textual content. This is a primary source of information on a file and is critical to driving correct behavior.

Textual content is typically what you would see were you to open a file in its native application. This includes not just body text, but visible text stored in other locations, such as headings, floating blocks, diagram subtitles, image captions, footnotes and endnotes, and other page furniture. Does the header say “Company Confidential”?

Tabular content can be particularly useful for downstream processing if it is identified as such. Table headers can give a vital clue as to the true meaning of cell contents, with “Social Security Number” making sense of an otherwise unlabeled string, avoiding an unmanageable number of false positive hits.

Hidden content is a significant part of many formats, containing valuable content beyond what is visible when viewed on-screen—such as explicitly hidden sections, tracked changes (including deleted text), cached content, scripts / macros, orphaned objects, accessibility content, and multiple representations for dynamic rendering based on viewer capabilities, for example using DOCX’s Alternate Content / Choice / Fallback features.

Container files embed additional files—subfiles. To get a complete view, these must be unpicked. Whilst the most obvious container files are archives, such as ZIP or RAR, there are many common non-archive file types which include subfiles, such as PDF (regular and Portfolio), Microsoft Office (OLE and XML format), and mailbox formats like PST. There is a wide range of standards for both container formats and compression codecs, and some container formats support additional components such as alternate data streams and different content based on the platform being used for extraction.

Most file formats are not publicly documented, and even for those that are, the published specification can be more useful as rough guidance than as fact. Implementations differ and have bugs, and truly understanding the nuances of what a format looks like in the wild will always require some level of reverse engineering.

*User-created content can be found in many places, a lot of which are not obvious. What are you missing? OpenText File Content Extraction is the high visibility option, exposing content that other solutions cannot reach.*

## Higher Performance

Maximize throughput, minimize latency, reduce CPU cost, decrease install size, optimize memory footprint—performance has many aspects. Designed to be a critical part of your document pipeline, OpenText File Content Extraction is never the bottleneck.

Lightning-fast accurate format detection performs everything necessary, but no more than what is required, to achieve a high confidence result. This can often be done on the first read, but sometimes deeper forensics are required for certainty—we know the difference.

Latency is reduced through output pipelining at a sub-document level, so time-critical downstream processing can start earlier. Once you find the information you need, such as the fact that there is PII in a document, you can terminate early and avoid the additional work in processing unnecessary parts of a file, reducing CPU usage.

Disk and memory usage can be a major consideration, particularly when deployed on an endpoint. OpenText File Content Extraction minimizes each of these, including residual memory when processing large files. Where certain functionality is not needed for a particular use case, your application can be further optimized by deploying OpenText File Content Extraction with a reduced footprint.

*When performance matters, choose a professionally maintained solution, proven by its deployment to millions of end users.*

## Designed for OEM

Unique applications have unique demands of the technology they embed. We understand this, providing a range of options to suit multiple deployment environments: endpoint, server/cloud, and hybrid models, with support for Windows, Linux and macOS, on both Intel and ARM architectures, with native APIs for a variety of programming languages.

Easily integrated libraries and reference code mean quick integration with new or existing applications. OpenText File Content Extraction supports both file-based and stream-based I/O for best fit with any application architecture.

Secure by design, OpenText File Content Extraction minimizes risk through techniques such as attack surface reduction for reduced threat impact, component isolation for quick third party vulnerability mitigation, process privilege reduction, and countermeasures for specific format-based attacks such as Zip Slip. A thread-safe, in-process or out-of-process model for threat containment, improves application stability in the face of any input data. We understand that one of the biggest security threats to your product is your supply chain.

Flexible licensing ensures that we can align with your business model.

OpenText File Content Extraction plays well with other embeddable OpenText technology such as OpenText™ Named Entity Recognition, enabling additional functionality and performance gains when used together.

*OpenText File Content Extraction exists to be embedded—we've been serving the OEM market for many years and understand the challenges you are facing.*

# Key Features

## File Format Detection

Reduce the risk of misprocessing crucial information or wasting valuable CPU time on irrelevant files by quickly and accurately identifying file type. Instead of relying exclusively on falsifiable file name extensions or short magic numbers, OpenText File Content Extraction forensically examines each file, focusing on the most differentiating characteristics first and going as deep as needed to resolve ambiguity, resulting in faster answers and a lower error rate.

OpenText File Content Extraction goes beyond MIME type, clearly identifying files with non-existent or ambiguous MIME types (e.g., application/octet-stream), adding detail such as encryption status, format classification, and format version, allowing you to make your downstream routing and processing decisions with precision.

Supported document classes include analytics, animation, CAD, database, desktop publishing, encapsulation, executable, font, GIS, library, movie, object module, outline, presentation graphics, raster image, schedule, scientific, sound, source code (including language identification), spreadsheet, vector graphics and word processing.

## Rights Management

Identify rights management protected files from Microsoft, Seclore and SmartCipher. Inspect MSIP (Microsoft Purview Information Protection) labels, even from encrypted files, to correctly determine risk.

Decrypt* files that have been protected by Microsoft Azure Rights Management (RMS), part of Microsoft Information Protection and associated technology, allowing your workflow to operate transparently on the original, unencrypted content.

## Metadata Access

Quickly access file metadata such as XMP, XrML, IPTC, EXIF, Boldon-James classification, and format specific fields. OpenText File Content Extraction combines and normalises common fields for easier downstream consumption.

## Character Set Conversion

Prepare for downstream processes, which usually expect UTF-8 input. OpenText File Content Extraction automatically determines the character set used within a document—even if this is not specified in the metadata—and converts this to UTF-8 or another encoding of your choice. With correct identification and conversion, value is maintained.

## Text Extraction

Extract plain text content by removing format scaffolding and other noise at speed.

Go deep into a wide variety of document formats, extracting body text and other visible components (such as headers, footers, footnotes, endnotes, captions and table components), with the option to include hidden text (such as section names, notes, tracked changes, explicitly hidden elements, accessibility layers and configurable placeholder text), as well as cached, orphaned, unused and deleted text.

* Credentials required

## Subfile Extraction

Dig into formats that commonly embed further content, from the obvious archive formats and email stores to more surprising container formats such as PDF and its variants, and all of the big three office-type documents: word processing, spreadsheet and presentation graphics.

## OCR

Directly access the textual content of scanned documents, photographed receipts, and raster images containing text, as part of the processing pipeline.

## Structured Export

Extract structure from document content, creating well-formed XML which is validated against a predefined Document Type Definition (DTD). OpenText File Content Extraction applies an XML vocabulary to the data structures in a document so that downstream applications can access content in context.

OpenText File Content Extraction returns structure such as headers / footers, footnotes, endnotes, bookmarks, headings, sheet names, and structured table data.

## HTML Export

Preview documents in high-fidelity HTML. Incorporating this technology into your web-based applications enables your end users to view a document even if they do not have the appropriate plug-in or native application.

With HTML Export, you control the content, structure, and format of the HTML output using either easily customized templates, or the flexible and robust APIs. Choose between web-friendly dynamically flowed text for best understanding of a document's content, or a fixed-width rendering, mimicking printed output.

Break content into manageable chunks for a faster load time and lower browser memory footprint. Add structure such as highlighting using custom markup, and automatically generate a navigable table of contents based on document properties such as font size or style. Apply Cascading Style Sheets (CSS) to improve output fidelity and align look and feel for a quick and easy read.

## PDF Export

Archive files to PDF format, ensuring that document content can be frozen and accessed for years to come without having to download specific apps to display each file type.

# System Requirements

| | APIs | | | | | Platforms | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | C | C++ | Python[1] | Java | .NET[2] | Windows/ x86_32 | Windows/ x86_64 | Windows/ ARM[3] | Linux/ x86_64 | Linux/ ARM | macOS/ x86_64 | macOS/ ARM |
| **File Format Detection** | • | • | • | • | • | • | • | • | • | • | • | • |
| **Rights Management[4]** | • | • | • | • | • | • | • | • | • | • | • | • |
| **Metadata Access[5]** | • | • | • | • | • | • | • | • | • | • | • | • |
| **Character Set Conversion** | • | • | • | • | • | • | • | • | • | • | • | • |
| **Text Extraction** | • | • | • | • | • | • | • | • | • | • | • | • |
| **Subfile Extraction** | • | • | • | • | • | • | • | • | • | • | • | • |
| **OCR** | • | • | • | • | • | | • | | • | • | | • |
| **Structured Export** | • | | | • | | • | • | | • | • | • | • |
| **HTML Export** | • | | | • | | • | • | | • | • | • | • |
| **PDF Export** | • | | | | | | • | | • | | | |

- Available

Learn more https://www.opentext.com/products/idol-keyview ›

1 Python available only on x86_64 platforms and macOS/ARM

2 .NET available only on Windows platforms

3 Windows/ARM supported only for C API

4 Decryption (separately licensable) available only in C and Java APIs, on Windows/x86_64 and Linux/x86_64 platforms

5 .NET metadata access excludes normalization



opentext™