

# All-in-One CLI Terminal File Converter

A robust, extensible Java-based command-line tool to convert files across formats with customizable options and automation support.

---

## Supported Input Formats (Current & Planned)

From	To	TXT	PDF	DOCX	ODT	HTML	CSV	JSON
TXT								
PDF								
DOCX								
ODT								
HTML								
CSV								
JSON								

Symbol	Meaning
Safe	Reliable, round-trip conversion possible
Lossy/Imperfect	May lose formatting, layout, or metadata
Unsupported	Not natively feasible or requires OCR/custom logic

### Documents

- TXT
- Planned: PDF, DOCX, ODT
- (Like `--text-only`, `--preserve-layout`, or `--ocr=true`)

Feature	Suggested Behavior
<code>--force</code>	Allow lossy conversions with warning
<code>--text-only</code>	Strip styles, useful for PDF → TXT, DOCX → TXT
<code>--ocr</code>	Enable scanned PDF to text (future feature)
<code>--preserve-layout</code>	Tries to maintain layout (e.g., HTML → DOCX)

•

### Images

- JPEG, PNG, BMP, TIFF
- Planned: GIF, SVG, WEBP, PSD (read-only)

### Audio

- Planned: MP3, WAV, FLAC, AAC, OGG (*via FFmpeg or similar*)

### Video

- Planned: MP4, AVI, MKV, MOV, FLV (*requires FFmpeg integration*)

## Archives

- Planned: ZIP, RAR, TAR, GZ, 7Z (*via Apache Commons Compress or 7z bindings*)

## Others

- Planned: CSV, JSON, XML, HTML (*via Jackson, Tika, jsoup, etc.*)
- 

## Output Format Capabilities

- Converts supported image formats (JPEG, PNG, BMP, TIFF, WEBP)
  - Planned: Cross-media conversions (e.g., TXT → PDF, DOCX → ODT, MP4 → MP3)
  - Autodetect input format and validate available output targets
  - Maintain metadata (EXIF for images, ID3 for audio)
- 

## Key Features

### User Interface

- Minimal CLI (`fconv`) with subcommands: `convert`, `formats`, `version`, `help`
- Optional flags: `--output-dir`, `--input`, `--to`
- Planned:
  - Real-time progress indicators (spinner, percent)
  - Batch file and directory conversion
  - Interactive shell mode

### Conversion Options

- Format autodetection via Apache Tika
  - Output path handling via centralized `OutputPathResolver`
  - Planned:
    - Image resizing, rotation, compression presets
    - Bitrate, codec, resolution controls for media
    - Page ranges for PDF (e.g., split, extract)
- 

## Integration and Extensibility

- Modular architecture: `FileConverter`, `ImageConversion`, `ConverterFactory`
- Plugin support (load converters via SPI or config)
- Planned:
  - CLI automation: pipeable input, script-friendly exit codes
  - API mode: expose converters as Java classes or REST services

- Self-update & plugin discovery
- 

## Performance and Reliability

- Graceful failure logging (invalid formats, file issues)
  - Future enhancements:
    - Multi-threaded batch processing
    - Memory and temp file cleanup
    - Retry/fallback logic for corrupt files
    - Integration with native tools like FFmpeg, ImageMagick for speed
- 

## Additional Utilities

- File metadata logging (via Tika + custom extractors)
  - Planned:
    - File encryption/decryption (AES, GPG)
    - Archive compression/decompression
    - Metadata editing for EXIF, ID3, PDF
    - Secure validation for hidden/suspicious file paths
- 

## Use Cases

- **Media professionals** converting large batches of audio/video/images
  - **Developers** automating data pipeline conversions (JSON → CSV, TXT → PDF)
  - **General users** managing files across devices or platforms (e.g., compress, standardize formats)
  - **Privacy-aware users** securely converting or encrypting sensitive documents
- 

## Next Milestones (Dev Notes)

Area	Action
Images	Add support for SVG, GIF read/write (ImageMagick, Batik)
Docs	Implement TXT → PDF, DOCX → ODT converters
/ Media	FFmpeg integration + CLI flags for codec/bitrate
CLI UX	Add <code>--overwrite</code> , <code>--verbose</code> , <code>--batch</code> , and interactive shell
Testing	Add unit tests for all converters + CLI integration tests
Extensibility	Register format handlers dynamically + future plugin system