Configuring Parsers At Parse Time in tika-server

With tika-server, Tika parsers can be configured at startup time via `tika-config.xml` OR at parse-time (per file) via headers.

NOTE: Parse-time/per file configuration was brought in slowly and was not well designed for extensibility (committers are standing by for a redesign!).

For the following reasons, when using Tika programmatically parse-time (per file) configurations must be set via the ParseContext for several reasons:

- Parsers may be loaded either via SPI or via the tika-config.xml – it is not easy to set configurations on the target parser.
- A given parser is likely wrapped in a CompositeParser and/or the AutoDetectParser, and it is not straightforward to be able to set parameters on the target parser.
- Parsers are used across threads so the configuration for a given file cannot be applied the parser generally.

Specifically, there are two popular config objects that are used in this way: the PDFParserConfig and TesseractOCRParserConfig. The parsers check for those configs in the ParseContext before parsing a file. These changes should only apply to the given file that is being parsed. These configs should not make changes to the configuration of the underlying parsers nor affect the parsing of other files.

Setting Parameters via Headers

As of February 2023 (version 2.7.0), Tika only directly supports configuration of the PDFParser and the TesseractOCRParser via headers. To set parameters for the PDFParser:

1) Create the header key by prepending `X-Tika-PDF` to a parameter, e.g. `X-Tika-PDFOcrStrategy`

2) Set the value (e.g. `ocr_only`)

To see the available parameters for the PDFParser, see: PDFParser (Apache PDFBox).

To set parameters for the TesseractOCRParser, prepend `X-Tika-OCR` to the parameter.

See the unit tests in the `tika-server-standard`'s `TikaResourceTest`, e.g. testOCRLanguageConfig().

Customizing Configuration

This functionality can be extended to other parsers by adding a class that implements ParseContextConfig and loading it via SPI. See the example of PDFServerConfig, and make sure to add your class via services, e.g. org.apache.tika.server.core.ParseContextConfig.