EPL 660: Lab 7
Apache Tika

University of Cyprus
Department of
Computer Science

Andreas Kamilaris

What is Apache Tika?

"Apache Tika is a toolkit for detecting and extracting metadata and structured text content from various documents using existing parser libraries."

from http://tika.apache.org/

Back in Time

- March 2007: Apache Tika project started.
- December 2007: Tika has made its first official release, titled 0.1-incubating.
- October 2008: Tika graduates to an Apache Lucene subproject.
- July 2009: Apache Tika 0.4 Release.
- February 2011: Apache Tika 0.9 Latest Release.

Supported Document Formats

- HyperText Markup Language
- XML and derived formats
- Microsoft Office document formats
- OpenDocument Format
- Portable Document Format
- Electronic Publication Format
- Rich Text Format
- Compression and packaging formats
- Text formats
- Audio formats
- Image& Video formats
- Java class files and archives
- The mbox format

Custom Parsers

University of Cyprus

- HTML
  - Through HtmlParser class.
  - The TagSoup library is used.
  - Supports virtually any kind of HTML found on the Web.
  - The output from the HtmlParser class is guaranteed to be well-formed and valid XHTML.
  - Various heuristics are used to prevent things like inline scripts from cluttering the extracted text content.

Custom Parsers

- XML
  - The Extensible Markup Language (XML) format is a generic format that can be used for all kinds of content.
  - Tika has custom parsers for some widely used XML vocabularies like XHTML, OOXML and ODF.
  - The default DcXMLParser class simply extracts the text content of the document and ignores any XML structure.

Custom Parsers

University of Cyprus

- Microsoft Office Document Formats
  - Microsoft Office and some related applications produce documents in the generic OLE 2 Compound Document and Office Open XML (OOXML) formats.
  - Since Office version 2007 the new XML-based OOXML format has been adopted.
  - The OfficeParser class supports text and metadata extraction from OLE2 documents.
  - The OOXMLParser class supports text and metadata extraction from OOXML documents.
  - Apache POI libraries are used.

Custom Parsers

University of Cyprus

- OpenDocument Format (ODF)
  - Used as the default format of the OpenOffice.org office suite.
  - The OpenDocumentParser class supports this format and the earlier OpenOffice 1.0 format on which ODF is based.
- Portable Document Format (PDF)
  - The PDFParser class parses PDF documents.
  - Apache PDFBox library is used.

Custom Parsers

- Rich Text Format (RTF)
  - The RTFParser class uses the standard javax.swing.text.rtf feature to extract text content from RTF documents.
- Compression and Packaging Formats
  - The PackageParser class (and its subclasses) first parse the top level compression or packaging format and then pass the unpacked document streams to a second parsing stage using the parser instance specified in the parse context.
  - Tika uses the Commons Compress library to support various compression and packaging formats.

Custom Parsers

University of Cyprus

- Text Formats
  - Extracting text content from plain text files seems like a simple task until you start thinking of all the possible character encodings.
  - The TXTParser class uses encoding detection code from the ICU project.
  - Automatic detection of the character encoding of a text document.

Custom Parsers

- Audio Formats
  - Tika can detect several common audio formats and extract metadata from them.
  - Even text extraction is supported for some audio files that contain lyrics or other textual content.
  - The AudioParser and MidiParser classes use standard javax.sound features to process simple audio formats.
  - The Mp3Parser class adds support for the widely used MP3 format.

Custom Parsers

University of Cyprus

- Image Formats
  - The ImageParser class uses the standard javax.imageio feature to extract simple metadata from image formats supported by the Java platform.
  - More complex image metadata is available through the JpegParser class that uses the metadata-extractor library to supports Exif metadata extraction from Jpeg images.

Custom Parsers

University of Cyprus

- Video Formats
  - Currently Tika only supports the Flash video format using a simple parsing algorithm implemented in the FLVParser class.
- Java Class Files and Archives
  - The ClassParser class extracts class names and method signatures from Java class files, and the ZipParser class supports also jar archives.
- The mbox Format
  - The MboxParser can extract email messages from the mbox format used by many email archives/mailboxes.

Content Detection

University of Cyprus

- The Detector Interface
  - The org.apache.tika.detect.Detector interface is the basis for most of the content type detection in Apache Tika.

```
MediaType detect (java.io.InputStream input,
                  Metadata metadata) throws java.io.IOException
```

  - The detector will return a MediaType object describing its best guess as to the type of the file.

- Mime Magic Detection
  - By looking for special patterns of bytes near the start of the file, it is often possible to detect the type of the file.

Content Detection

- Resource Name Based Detection
  - When the name of the file is known, it is sometimes possible to guess the file type from the name or extension. Within the tika-mimetypes.xml file is a list of patterns which are used to identify the type from the filename.

- Known Content Type Detection
  - Sometimes, the mime type for a file is already known, such as when downloading from a Web server, or when retrieving from a content store. This information can be used by detectors.

Default Content Detection

University of Cyprus

- The default Mime Types Detector
  - By default, the mime type detection in Tika is provided by org.apache.tika.mime.MimeTypes.
  - This detector makes use of tika-mimetypes.xml to power magic-based and filename-based detection.
  - The procedure is:
    - Magic-based detection is used on the start of the file.
    - If the file is an XML file, then the start of the XML is processed for root elements.
    - If available, the filename is then used to improve the detail of the detection.
    - Finally, if available, the supplied content type is used to further refine the type.

Container Aware Detection

- Several common file formats are actually held within a common container format. Using magic detection, you can spot that a given file is e.g. an OLE2 document.
- When speed is important, learning quickly the container type is sufficient. For other cases, you need to process the container to get an accurate answer on its contents. For these cases, a container aware detector should be used.
- Tika provides a wrapping detector in the parsers bundle, called org.apache.tika.detect.ContainerAwareDetector. This detector will check for certain known containers, and if found, will open them and detect the appropriate type based on the contents.

Language Detection

University of Cyprus

- Tika is able to help identify the language of a piece of text, which is useful when extracting text from document formats which do not include language information in their metadata.
- The language detection is provided by org.apache.tika.language.LanguageIdentifier.

Parser Characteristics

University of Cyprus

- The main characteristics of the Tika Parser are:
  - Streamed parsing.
  - Structured content (e.g. headings, links in the extracted content).
  - Input metadata (filename, content type).
  - Output metadata (return document metadata in addition to document content).
  - Context sensitivity (more fine-grained control over the parsing process).

Parser Interface

- The org.apache.tika.parser.Parser interface is the key concept of Apache Tika.
- It hides the complexity of different file formats and parsing libraries while providing a simple and powerful mechanism for client applications to extract structured text content and metadata from all sorts of documents.

```
void parse( InputStream stream, ContentHandler handler,
                Metadata metadata, ParseContext context)
            throws IOException, SAXException, TikaException;
```

- The first argument is an InputStream for reading the document to be parsed. The ParseContext argument is used to inject context-specific information to the parsing process.
- The parse method takes the document to be parsed and related metadata as input and outputs the results as XHTML SAX events and extra metadata.

Parser Interface

University of Cyprus

- The parsed content of the input stream is returned as a sequence of XHTML SAX events. XHTML is used to express structured content of the document and SAX events enable streamed processing.
- The XHTML SAX events, produced by the parser implementation, are sent to a ContentHandler instance.
- The overall structure of the generated event stream is:

```
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>...</title>
  </head>
  <body>
    ...
  </body>
</html>
```

Document Metadata

University of Cyprus

- The more interesting metadata properties are:
  - Metadata.RESOURCE_NAME_KEY (the name of the file or resource that contains the document).
  - Metadata.CONTENT_TYPE (the declared content type of the document).
  - Metadata.TITLE
  - Metadata.AUTHOR

Getting Started with Apache Tika

1. Download a source release or checkout the latest sources from version control: http://tika.apache.org/download.html
2. Build Tika from sources: use Maven 2 build system
   (sudo apt-get install maven2; mvn install)
3. The Tika 0.9 build consists of a number of components and produces the following main binaries:
   - tika-core/target/tika-core-0.9.jar
     - Tika core library. Contains the core interfaces and classes of Tika, but none of the parser implementations. Depends only on Java 5.
   - tika-parsers/target/tika-parsers-0.9.jar
     - Tika parsers. Collection of classes that implement the Tika Parser interface based on various external parser libraries.
   - tika-app/target/tika-app-0.9.jar
     - Tika application. Combines the above libraries and all the external parser libraries into a single runnable jar with a GUI and a command line interface.
   - tika-bundle/target/tika-bundle-0.9.jar
     - Tika bundle. An OSGi bundle that includes everything you need to use all Tika functionality in an OSGi environment.

Using Apache Tika

4. There exist different ways to use Apache Tika:
   - As a Maven dependency.
   - In an Ant project: Include the Tika jar files and the dependencies individually.
   - As a command line utility: The Tika application jar (tika-app-0.9.jar) can be used for extracting text content and metadata from all sorts of files.

Tika As a Command Line Utility

University of Cyprus

- The usage instructions to run Tika as a command line utility are:

usage: java -jar tika-app-0.9.jar [option] [file]

Options:

-? or --help Print this usage message

-v or --verbose Print debug level messages

-g or --gui Start the Apache Tika GUI

-x or --xml Output XHTML content (default)

-h or --html Output HTML content

-t or --text Output plain text content

-m or --metadata Output only metadata

Description:

Apache Tika will parse the file(s) specified on the command line and output the extracted text content or metadata to standard output.

Instead of a file name you can also specify the URL of a document to be parsed. If no file name or URL is specified (or the special name "-" is used), then the standard input stream is parsed.

Use the "--gui" (or "-g") option to start the Apache Tika GUI. You can drag and drop files from a normal file explorer to the GUI window to extract text content and metadata from the files.

You can also use the jar as a component in a Unix pipeline or as an external tool in many scripting languages.

Create your own Parser

1. Add your MIME-Type:
   - You first need to modify tika-core/src/main/resources/org/apache/tika/mime/tika-mimetypes.xml in order to Tika can map the file extension with its MIME-Type.

```
<mime-type type="application/hello">
      <glob pattern="*.hi"/>
</mime-type>
```

Create your own Parser

University of Cyprus

2. Create your Parser class:
   - This is a class that must implement the Parser interface offered by Tika.

```
public class ExampleParser implements Parser {

        public void parse( InputStream stream, ContentHandler handler,
                           Metadata metadata, ParseContext context)
                    throws IOException, SAXException, TikaException {

            metadata.set(Metadata.CONTENT_TYPE, HELLO_MIME_TYPE);
            metadata.set("Hello", "World");
            XHTMLContentHandler xhtml = new HTMLContentHandler(handler,
                                                               metadata);
            xhtml.startDocument();
            xhtml.endDocument();
        }
}
```

Live Demonstration

University of Cyprus

- Parse some document with Apache Tika (Ant project).
- Learn utilizing custom parsers.
- In this demo:
  - parse a pdf file with PDFParser class.
  - Parse an xml file with DcXMLParser class

```
void parse_pdf_document(String path_pdf_file){

        InputStream input = new FileInputStream(new File(path_pdf_file));
        ContentHandler textHandler = new BodyContentHandler();
        Metadata metadata = new Metadata();
        PDFParser parser = new PDFParser();
        parser.parse(input, textHandler, metadata);
        input.close();
        System.out.println("File contents: " + textHandler.toString());
}
```

Location of the pdf file

Parser for pdf files

Define any needed metadata

Print parsed content

Useful Info

University of Cyprus

- Official Apache Tika site: http://tika.apache.org/
- Tika Wiki: http://wiki.apache.org/tika/
- Tika API Documentation: http://tika.apache.org/0.9/api/
- Tika Tutorial: http://www.ibm.com/developerworks/opensource/tutorials/os-apache-tika/index.html
- Apache Tika Presentation: http://www.slideshare.net/paolomoz/content-analysis-with-apache-tika-presentation