
Chris A. Mattmann, Ph.D.
Jet Propulsion Laboratory, California Institute of Technology /
University of Southern California / Apache Software Foundation

# Content Detection and Analysis for Big Data

Sponsored by the Jet Propulsion Laboratory and by the Keck Institute for Space Studies, California Institute of Technology

# Outline

- The Information Landscape
- Importance of Content Detection
- Challenges
- Approaches
- Introduction to Apache Tika
- Using Apache Tika
- Wrap-up

# Introduction to Apache Tika

# Outline

- What is Tika?
- Where did it come from?
- What are the current versions of Tika?
- What can it do?

# Apache Tika is…

- A content analysis and detection toolkit
- A set of Java APIs providing MIME type detection, language identification, integration of various parsing libraries
- A rich Metadata API for representing different Metadata models
- A command line interface to the underlying Java code
- A GUI interface to the Java code

# Tika's (Brief) History

- Original idea for Tika came from Chris Mattmann and Jerome Charron in 2006
- Proposed as Lucene sub-project
  - Others interested, didn't gain much traction
- Went the Incubator route in 2007 when Jukka Zitting found that there was a need for Tika capabilities in Apache Jackrabbit
  - A Content Management System
- Graduated from the Incubator to Lucene sub-project in 2008
- Graduated to Apache TLP in 2010

# Getting started rapidly

- Download Tika from:
  - http://tika.apache.org/download.html
- Grab tika-app-1.5.jar
- alias tika “java –jar tika-app-1.5.jar”
- tika < somefile.doc > extracted-text.xhtml
- tika –m < somefile.doc > extracted.met

# Detecting MIME types from Java

- String type = Tika.detect(…)
  - java.io.InputStream
  - java.io.File
  - java.net.URL
  - java.lang.String

# Adding new MIME types

- Got XML?

```
– <mime-info>
    <mime-type type="application/activemessage"/>
  – <mime-type type="application/andrew-inset">
      <glob pattern="*.ez"/>
    </mime-type>
    <mime-type type="application/applefile"/>
  – <mime-type type="application/applixware">
      <glob pattern="*.aw"/>
    </mime-type>
  – <mime-type type="application/atom+xml">
      <root-XML localName="feed" namespaceURI="http://purl.org/atom/ns#"/>
      <glob pattern="*.atom"/>
    </mime-type>
```

# Parsing

- String content = Tika.parseToString(…)
  - InputStream
  - File
  - URL

# Streaming Parsing

- Reader reader = Tika.parse(…)
  - InputStream
  - File
  - URL

# Language Detection

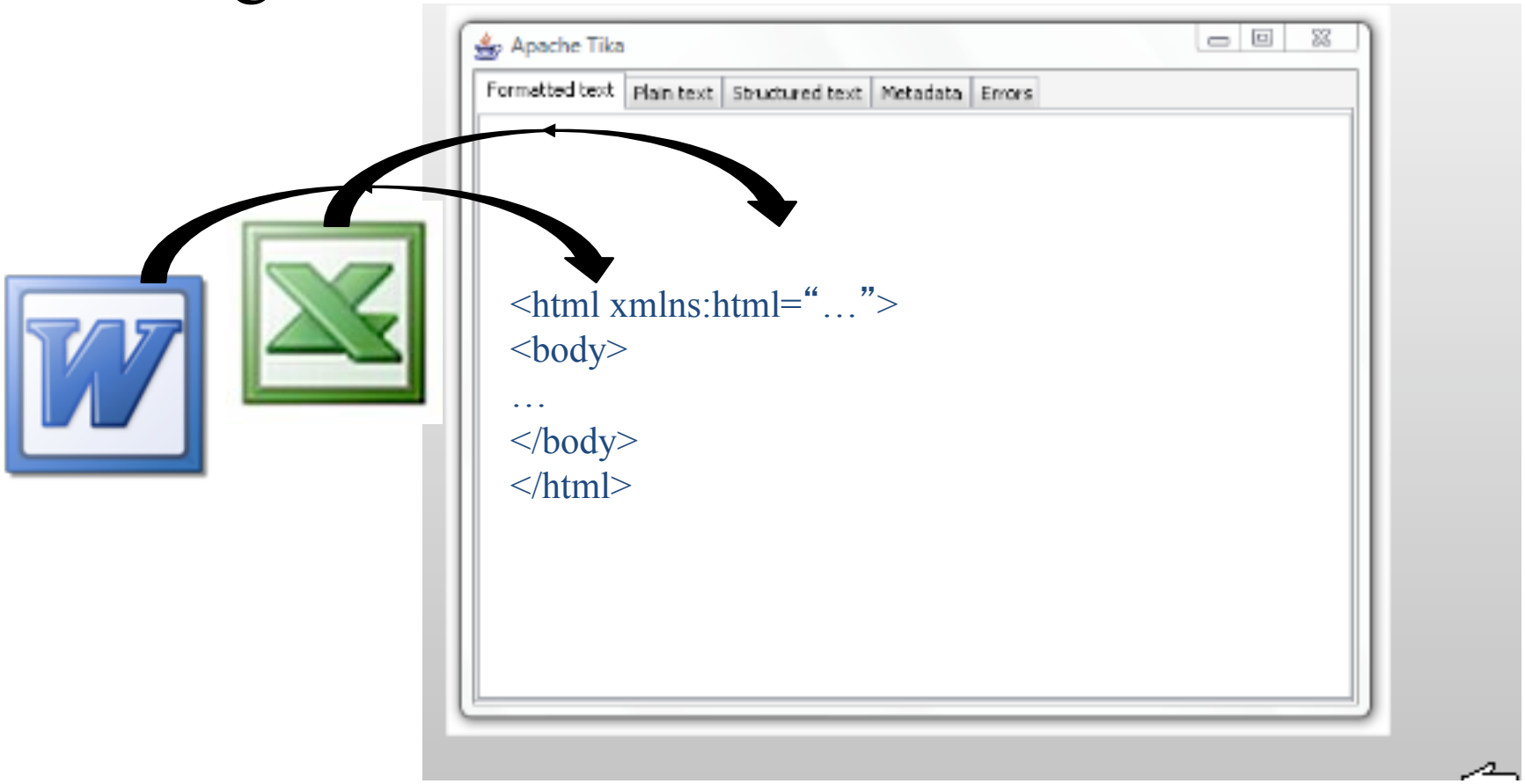
- LanguageIdentifier lang = new LanguageIdentifier(new LanguageProfile( FileUtils.readFileToString(new File(filename))));
- System.out.println(lang.getLanguage());
- Uses Ngram analysis included with Tika
  - Originating from Nutch
  - Can be improved

# Metadata

- Metadata met = new Metadata();
  //Dubiln Core
  met.set(Metadata.FORMAT, “text/html”);
  //multi-valued
  met.set(Metadata.FORMAT, “text/plain”);
  System.out.println(
  met.getValues(Metadata.FORMAT));
- Other met models supported (HTTP Headers, Word, Creative Commons, Climate Forcast, etc.)

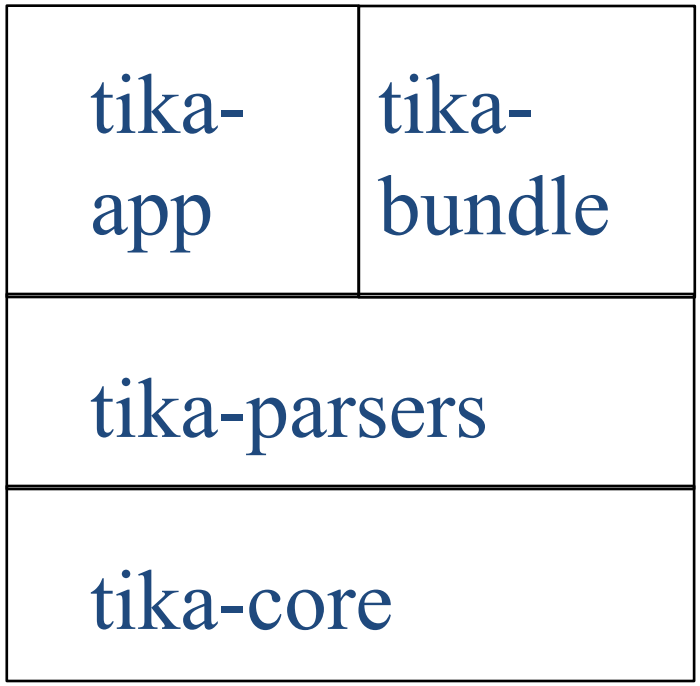
# Running Tika in GUI form

- tika --gui



# Integrating Tika into your App

- Maven
- Ant
- Eclipse
- It's just a set of jars
  - tika-core
  - tika-parsers
  - tika-app
  - tika-bundle



# Wrapup

- Lots more information at
  - http://tika.apache.org
- Possible projects
  - Adding more parsers for content types
    - Omnigraffle?
  - Expanding ability to handle random access file parsing
    - Scientific data file formats
  - Improving language and charset detection
  - Machine Translation – add new Translation APIs

# Acknowledgements


- Material inspired by Jukka Zitting's talks
  - http://www.slideshare.net/jukka/text-and-metadata-extraction-with-apache-tika
  - http://www.slideshare.net/jukka/text-and-metadata-extraction-with-apache-tika-4427630

# Some Pointers

- CSCI 572: Search Engines at USC
  - http://www-scf.usc.edu/~csci572/
- Mattmann Home Page
  - http://sunset.usc.edu/~mattmann/
- Tika in Action
  - http://manning.com/mattmann/
- Apache Tika
  - http://tika.apache.org/