So what are we covering?

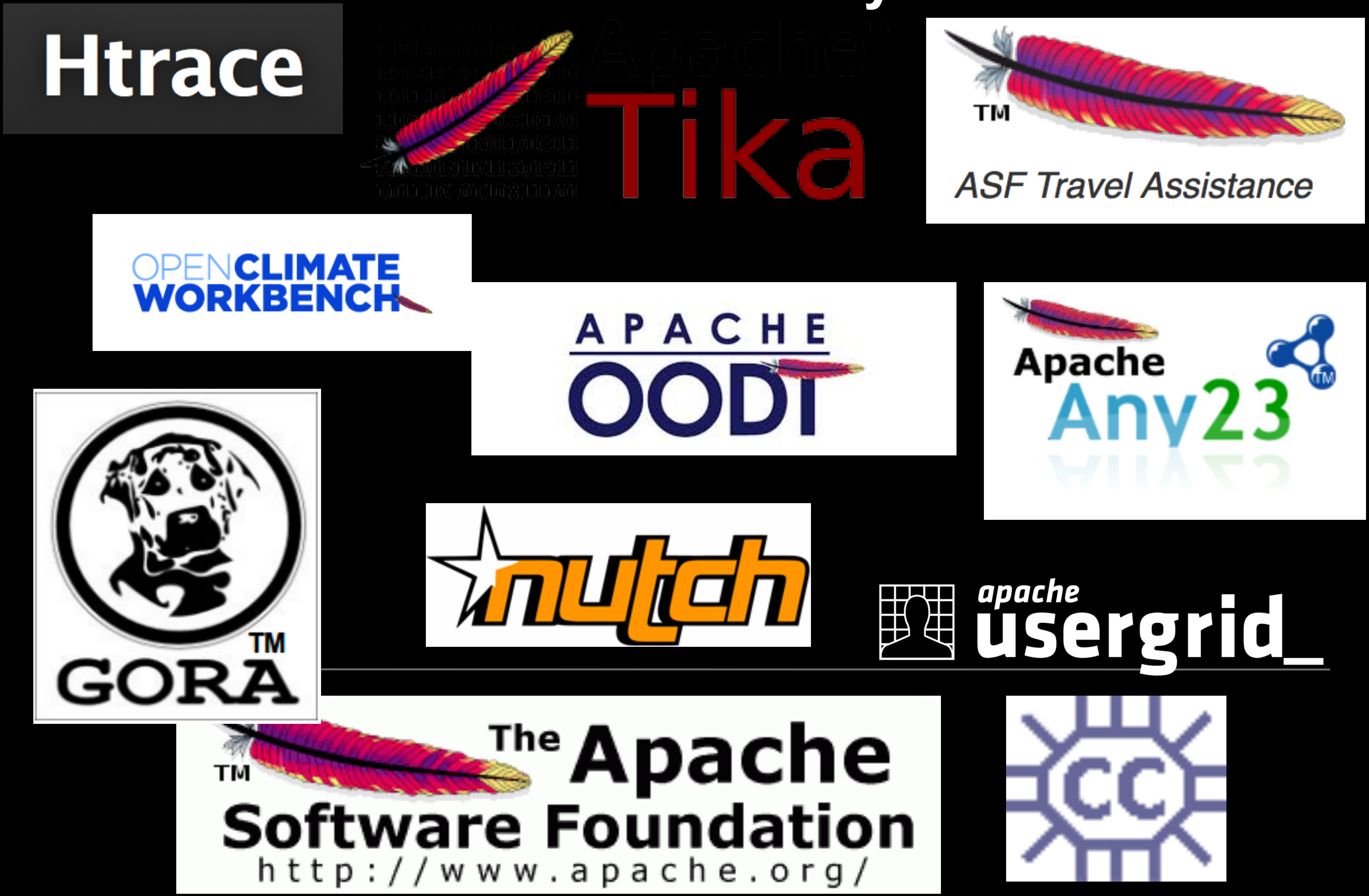
- Me, Myself and I + Apache
- Contextual motivation for improved i18n… and i18n services
- The Apache Tika.translate API
- PO.DAAC
- The iPReS Project
- Demo iPReS Web Service
- Discussion on next steps, limitations and a home for iPReS
- Conclusion and recap

★★★★
"RIVETINGLY ENJOYABLE"
"HEART POUNDING... BRILLIANT"
"A STUNNING ACHIEVEMENT"
"SENSATIONAL"
ACADEMY AWARD WINNING
KEVIN MACDONALD
THE LAST KING OF SCOTLAND
FOREST WHITAKER
JAMES McAVOY
KERRY WASHINGTON

Many hats for many occasions

TM The Apache Software Foundation http://www.apache.org/

How much is many?



Contextual motivation for improved i18n… specifically i18n services

So why Internationalization… now?

Summer 2014: Involvement as performer on DARPA's XDATA Program (PI Chris Mattmann).

DARPA provide a number of datasets such as

- Employment opportunities posted from http://www.computrabajo.com affiliate sites for Mexico and South American countries. Postings are temporary and may be taken down at any time due to a number of factors so this data set is an attempted persistence of these postings for analysis over a long period of time.
- Netscan tracing results of three different types of distributed scans across the internets IPv4 address speace over a period of time. Collected from many 100,000s different machines. Containing info such as IP address, scan ts scan result, HTTP response status codes
- Web Data Commons one of the largest web page hyperlink graphs available to the public outside of companies such as Google, Yahoo, and Microsoft. Extracted from CommonCrawl (which uses Apache Nutch)
- NBA Game Recap Dataset consists of two parts: **1)** Structured game log data dating back to 2010-2011 season including player statistics, scores, play-by-play events, and other metadata and **2)** Unstructured game recap text and message board comments associated with the structured data. The linkages of these two data sets provide for a wide range of unstructured text analytics against a backdrop of game result ground truth.

Employment Dataset Characteristics

- 119+ M jobs postings
- 40GB
- Approximately 2.1 M unique job postings… many duplicates
- … loads of other specifics
- The Translated Location field (NOT using Apache Tika) was parsed out from the data and run through a geo-fixing service to estimate a rough latitude and longitude
- It was quickly discovered, when job postings were located as being presenting in the mid Indian Ocean, that there were discrepancies in the geo-location characteristics.

!!!REGARDLESS!!!

THE ENTIRE DATASET IS IN SPANISH

Data Field	Example
Posted Date	2012-10-23
Location	Capital Federal
Department	Capital Federal
Title	Desarrollador plataforma SalesForce CRM.
Salary	A convenir
Start	Inmediato
Duration	Indeterminada
Job Type	Tiempo Completo
Applications	Enviar Cv con Ref Desarrollador SalesForce CRM
Company	Softtek
Contact	Belen Lavinia
Phone	
Fax	
Translated Location	Buenos Aires, Argentina
Latitude	-34.6037232
Longitude	-58.3815931
Date First Seen	2012-10-29
URL	http://www.computrabajo.com.ar/bt-ofrd-softtek-21444.htm
Date Last Seen	2012-11-06

Table 1: Employment Data Fields

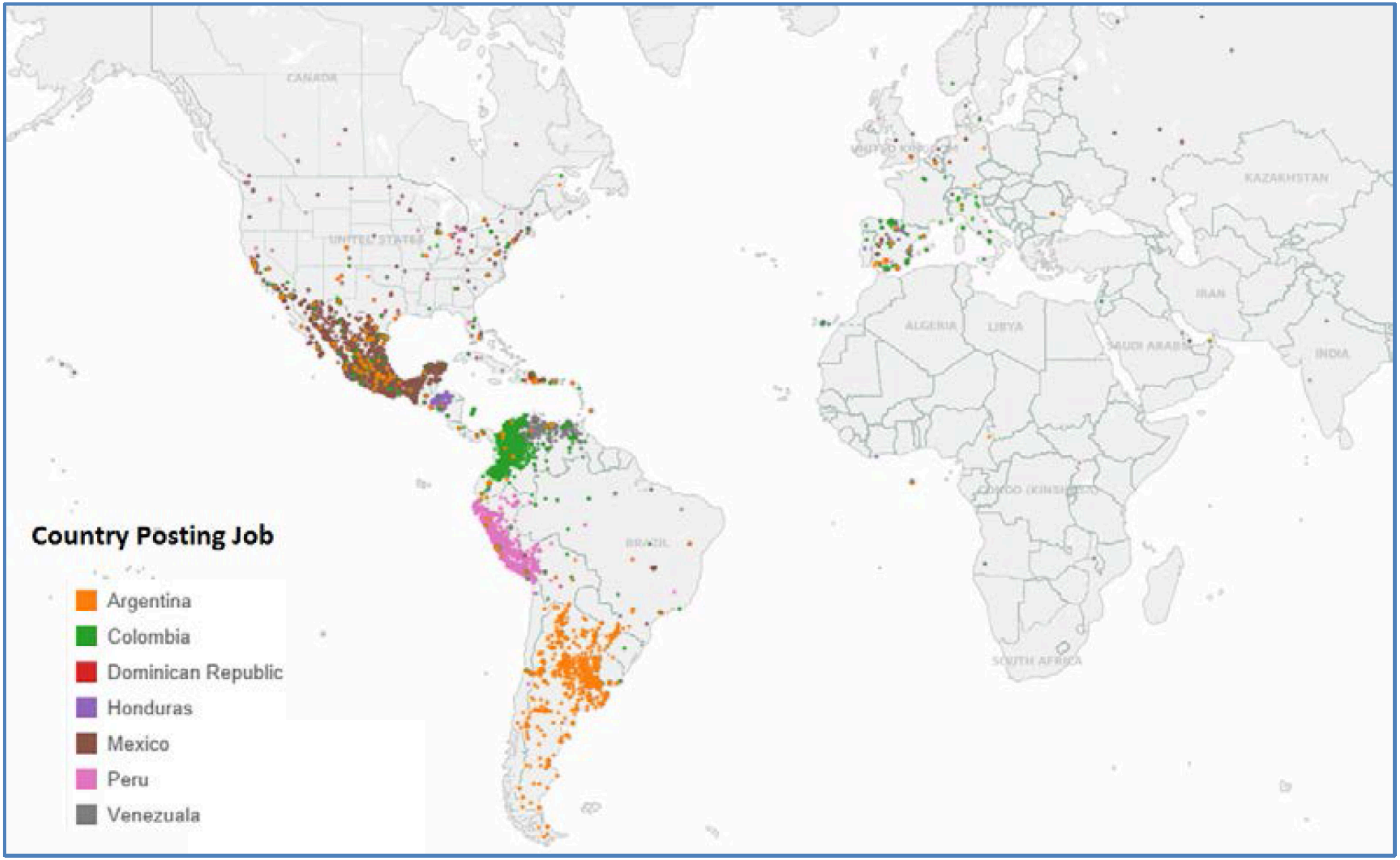


Figure 1: Map of Jobs (Colored by Country)

Example Employment Challenges

- Predict which geospatial areas will have which job types in the future
- Predict how long job postings will exist based on job type
- Discover temporal or geospatial trends or anomalies in job postings. Can you find events which correlate to the localized job offerings?
- Join job URL's with WDC Hyperlinks, Akamai dara, and/or Net Scan data to find affiliations and interesting observations. Benchmarking joining processes.
- … and so forth

Oh yeah, and did I mention the dataset is in Spanish? Yes I did!

Queue Tika.translate

Example Employment Challenges

Predict which geospatial areas will have which job types in the future

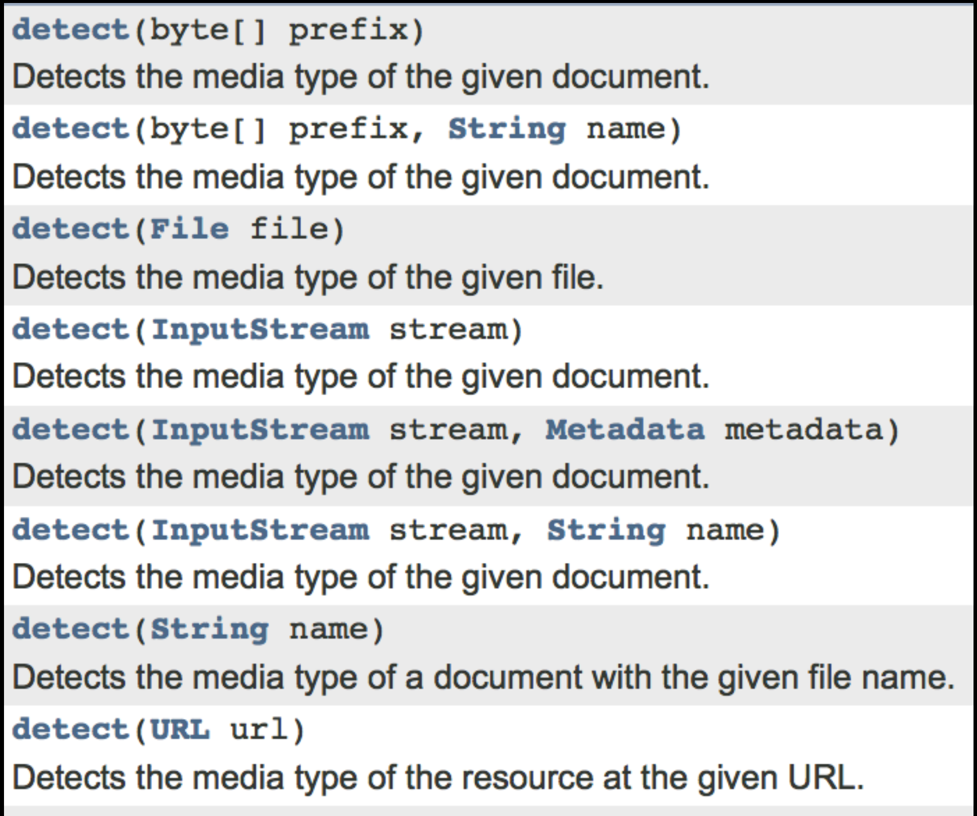
Predict how long job postings will exist based on job type

Join job URL's with WDC Hyperlinks, Akamai data, and/or Net Scan data to find affiliations and interesting observations. Benchmarking joining processes.

Queue Tika.translate

The Tika.translate addition to Tika API

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



`detect(byte[] prefix)`
Detects the media type of the given document.

`detect(byte[] prefix, String name)`
Detects the media type of the given document.

`detect(File file)`
Detects the media type of the given file.

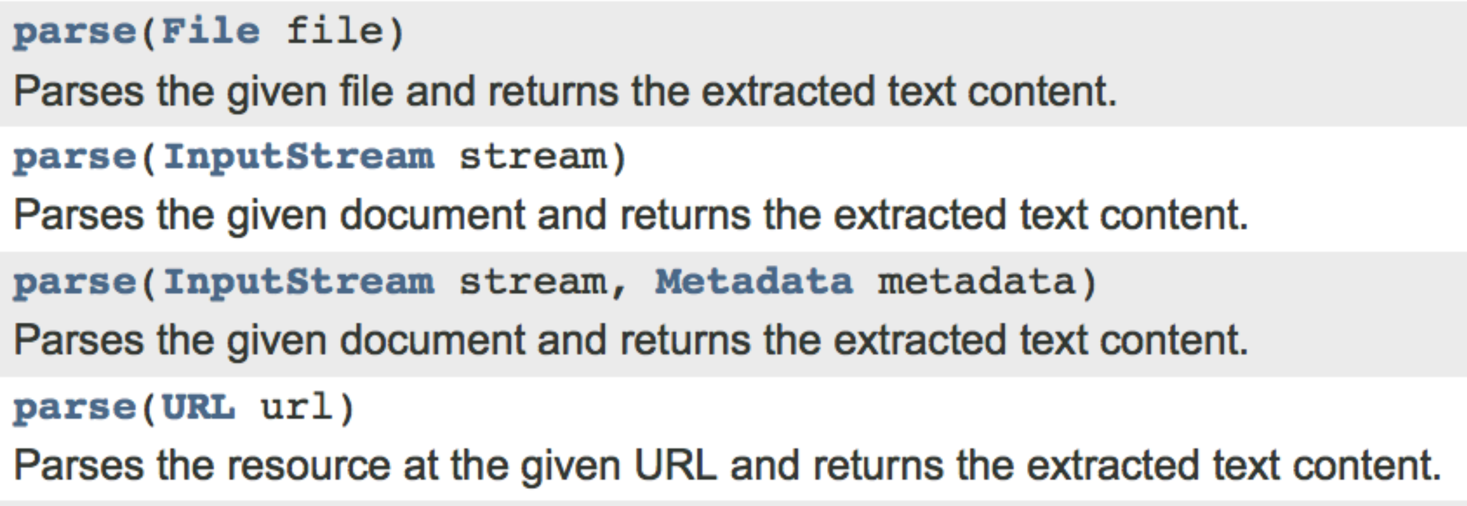
`detect(InputStream stream)`
Detects the media type of the given document.

`detect(InputStream stream, Metadata metadata)`
Detects the media type of the given document.

`detect(InputStream stream, String name)`
Detects the media type of the given document.

`detect(String name)`
Detects the media type of a document with the given file name.

`detect(URL url)`
Detects the media type of the resource at the given URL.



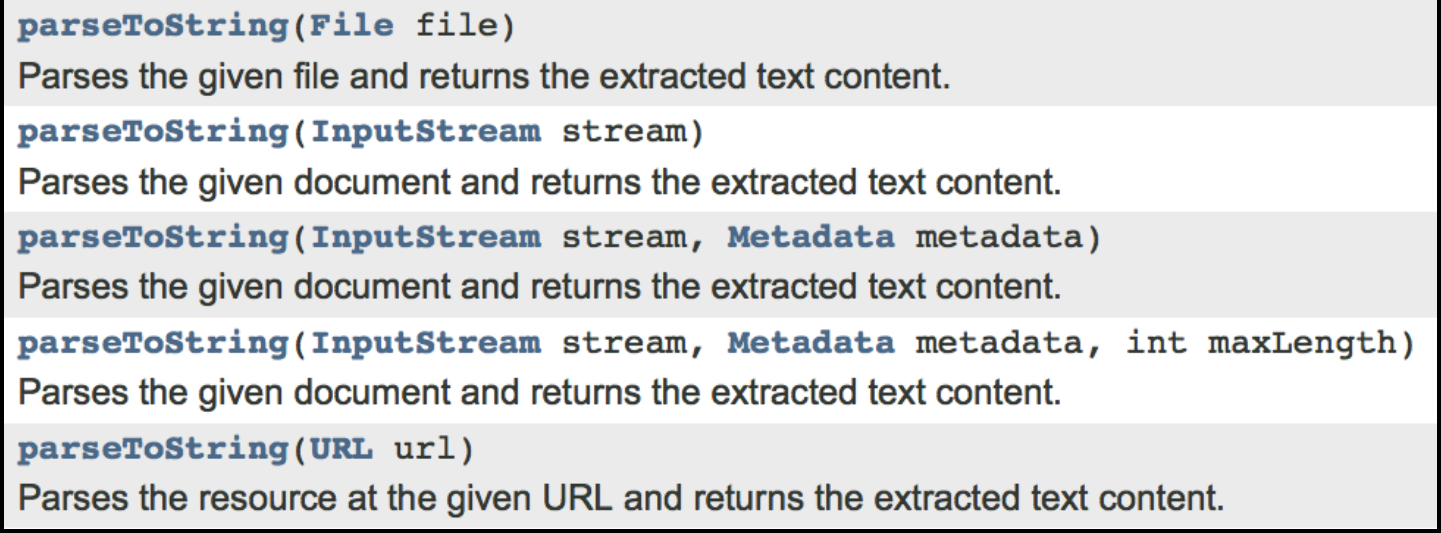
`parse(File file)`
Parses the given file and returns the extracted text content.

`parse(InputStream stream)`
Parses the given document and returns the extracted text content.

`parse(InputStream stream, Metadata metadata)`
Parses the given document and returns the extracted text content.

`parse(URL url)`
Parses the resource at the given URL and returns the extracted text content.

Apache Tika API Cont'd



parseToString(**File** file)

Parses the given file and returns the extracted text content.

parseToString(**InputStream** stream)

Parses the given document and returns the extracted text content.

parseToString(**InputStream** stream, **Metadata** metadata)

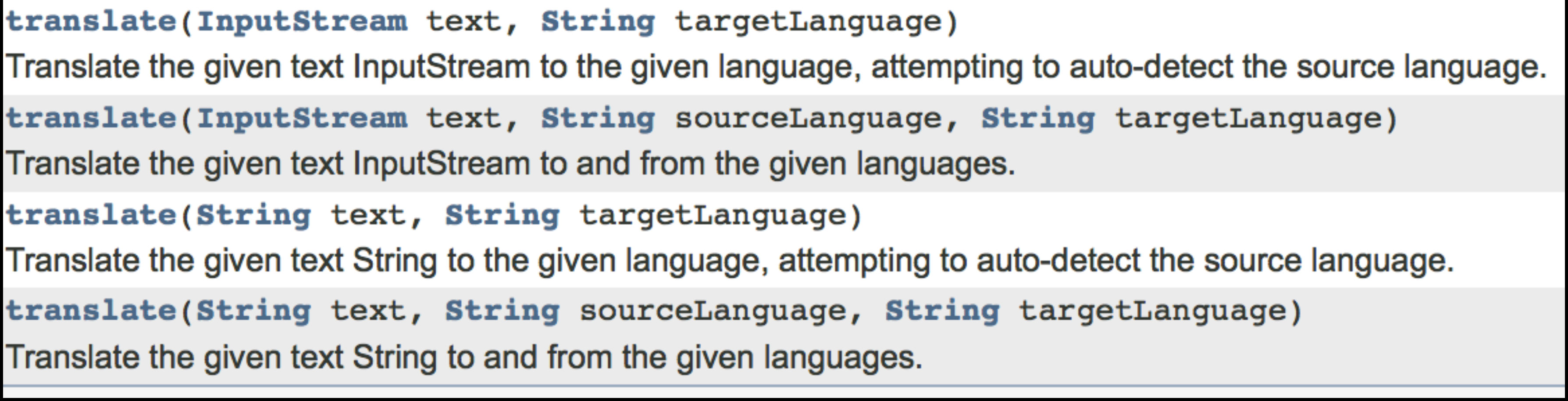
Parses the given document and returns the extracted text content.

parseToString(**InputStream** stream, **Metadata** metadata, int maxLength)

Parses the given document and returns the extracted text content.

parseToString(**URL** url)

Parses the resource at the given URL and returns the extracted text content.



translate(**InputStream** text, **String** targetLanguage)

Translate the given text InputStream to the given language, attempting to auto-detect the source language.

translate(**InputStream** text, **String** sourceLanguage, **String** targetLanguage)

Translate the given text InputStream to and from the given languages.

translate(**String** text, **String** targetLanguage)

Translate the given text String to the given language, attempting to auto-detect the source language.

translate(**String** text, **String** sourceLanguage, **String** targetLanguage)

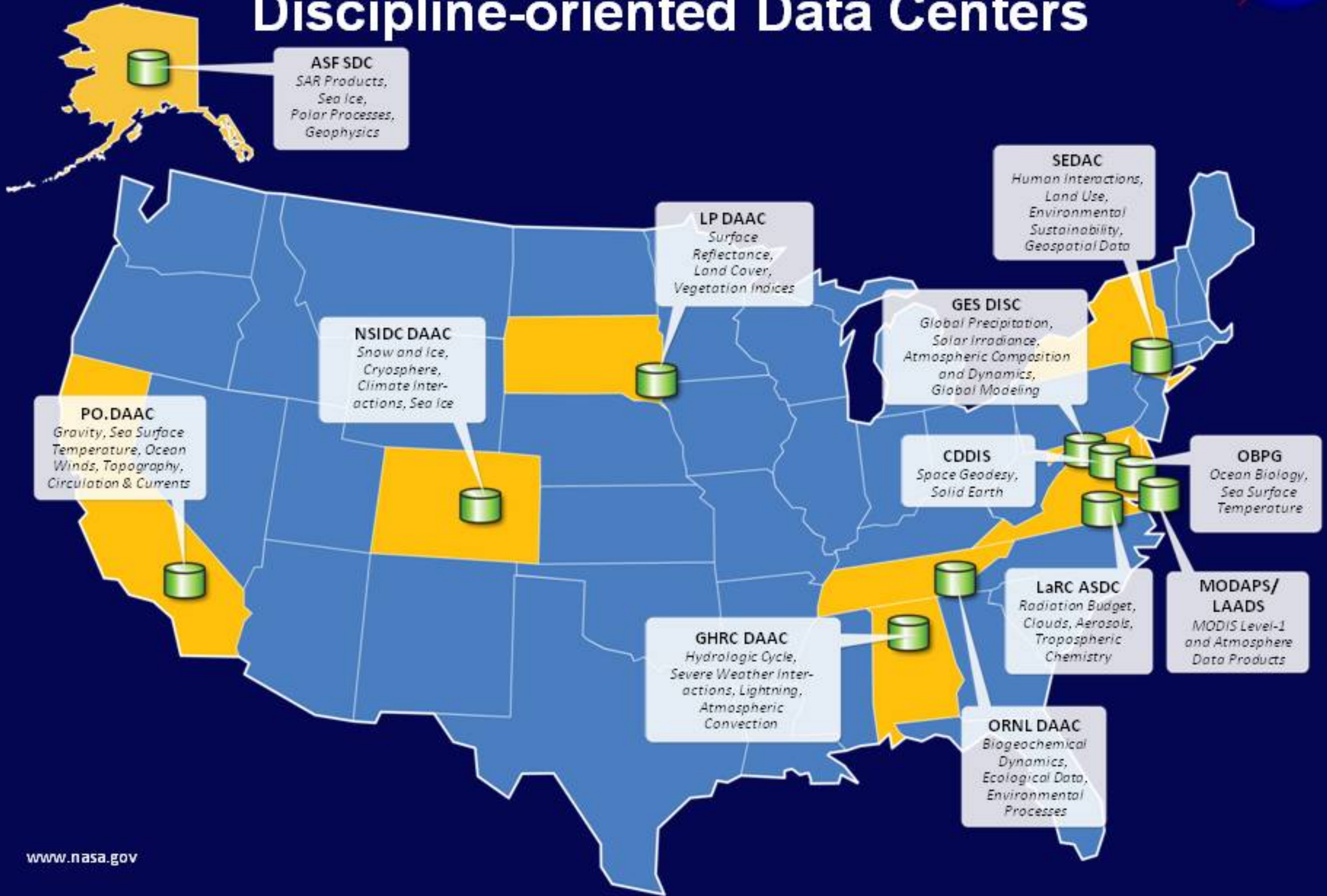
Translate the given text String to and from the given languages.

Added module and core Tika interface for translating text between languages and added a default implementation that call's Microsoft's translate service (TIKA-1319)

NASA JPL's Physical Oceanographic Data Active Archive Centre… otherwise known as PO.DAAC

NASA
Jet Propulsion Laboratory
California Institute of Technology
podaac
Physical Oceanography Distributed Active Archive Center

Discipline-oriented Data Centers



ASF SDC
*SAR Products,
Sea Ice,
Polar Processes,
Geophysics*

SEDAC
*Human Interactions,
Land Use,
Environmental
Sustainability,
Geospatial Data*

LP DAAC
*Surface
Reflectance,
Land Cover,
Vegetation Indices*

GES DISC
*Global Precipitation,
Solar Irradiance,
Atmospheric Composition
and Dynamics,
Global Modeling*

NSIDC DAAC
*Snow and Ice,
Cryosphere,
Climate Inter-
actions, Sea Ice*

PO.DAAC
*Gravity, Sea Surface
Temperature, Ocean
Winds, Topography,
Circulation & Currents*

CDDIS
*Space Geodesy,
Solid Earth*

OBPG
*Ocean Biology,
Sea Surface
Temperature*

LaRC ASDC
*Radiation Budget,
Clouds, Aerosols,
Tropospheric
Chemistry*

**MODAPS/
LAADS**
*MODIS Level-1
and Atmosphere
Data Products*

GHRC DAAC
*Hydrologic Cycle,
Severe Weather Inter-
actions, Lightning,
Atmospheric
Convection*

ORNL DAAC
*Biogeochemical
Dynamics,
Ecological Data,
Environmental
Processes*

- Distribution of data for sea surface temperature, sea surface topography, and ocean vector winds acquired by NASA instruments.
- Petabytes of Data… heterogeneous data products e.g. array-based (netCDF3, 4, HDF4/5), Binary Data Products, TIFF, GeoTIFF, etc.
- The primary goal (and challenge) for PO.DAAC is to enable provision, dissemination and availability of such data to the global scientific community at large.

The iPReS Project
Internationalization Product Retrieval Service

iPReS in a Nutshell

The Internationalization (i18n) Product Retrieval Service is a web service and client providing i18n-type access to products and product metadata contained within NASA JPL Physical Oceanography Distributed Active Archive Center otherwise known as PO.DAAC.

The software implements a RESTful PO.DAAC Web-Services API.

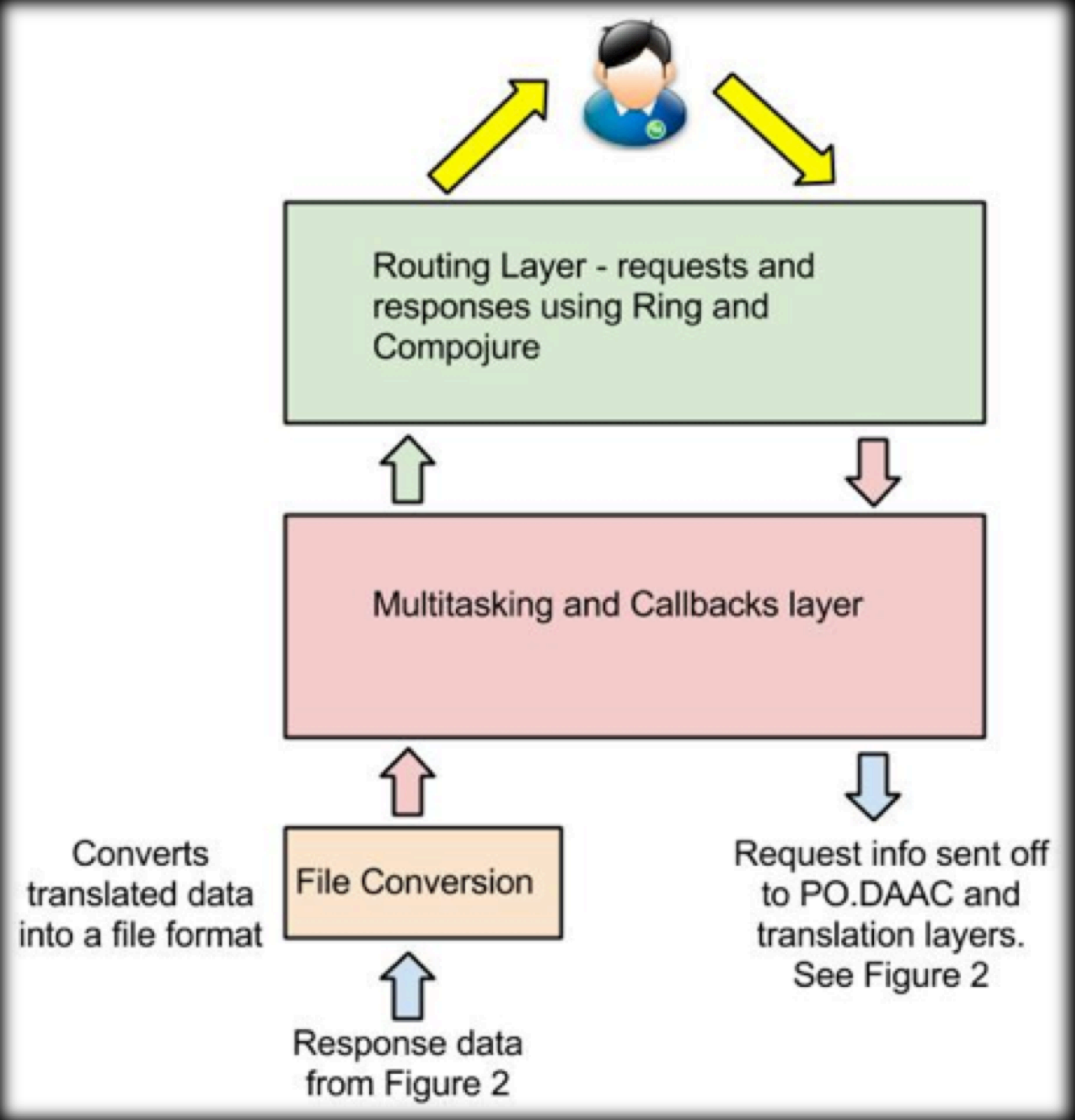
It then leverages the Tika.translate API to translate scientific product metadata into a target language provided along with the initial call to the service.

Project Characteristics

- Initially proposed and accepted as a Capstone project in August 2014 based on Steve Hathaway posting notification to community@
- Three Oregon State University students, Phillip Carter, Bhavik Vikram Patel and Daniel Song 20% of CS Masters degree.
- 6 month project…

http://lewismc.github.io/iPReS/

Design and Architecture



Routing Layer - requests and responses using Ring and Compojure

Multitasking and Callbacks layer

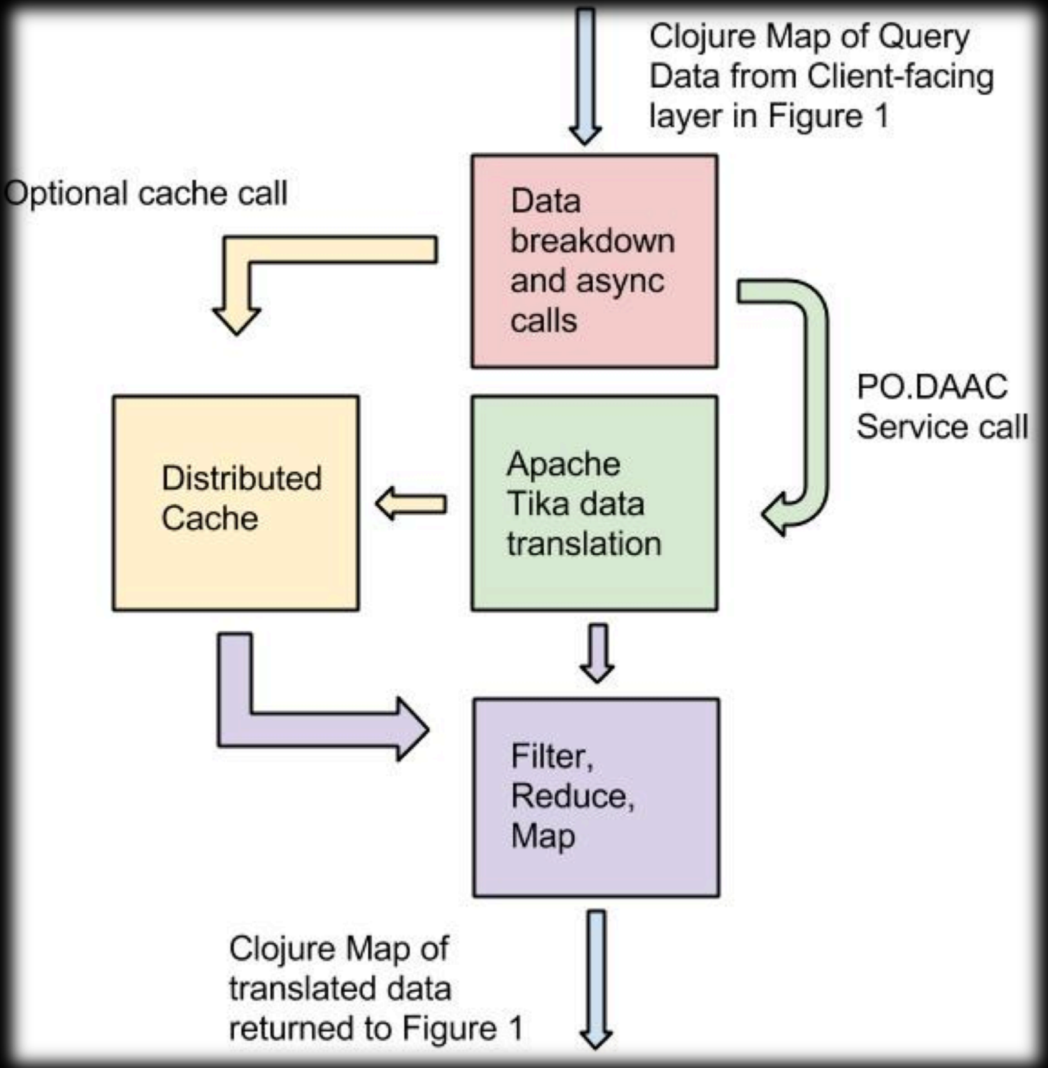
Converts translated data into a file format

File Conversion

Request info sent off to PO.DAAC and translation layers. See Figure 2

Response data from Figure 2

Design and Architecture Cont’d


Clojure Map of Query Data from Client-facing layer in Figure 1
Optional cache call
Data breakdown and async calls
PO.DAAC Service call
Distributed Cache
Apache Tika data translation
Filter, Reduce, Map
Clojure Map of translated data returned to Figure 1

iPReS Demo

Discussion on next steps, limitations and a home for iPReS

Already Licensed under ALv2.0… obviously

Apache Incubator not the right place however PO.DAAC Labs maybe is!

Low Technology Readiness Level (TRL) … collaborate with other parties to further develop the concept for federated i18n search across other NASA DAAC’s.

iPReSaaS @NASA JPL

TIKA-1343 **Create a Tika Translator implementation that uses JoshuaDecoder**

Conclusion and Recap

What did we cover?

- Contextual motivation for improved I8n… and I8n services
- The Apache Tika.translate API
- PO.DAAC
- The iPReS Project
- Demo iPReS Web Service
- Discussion on next steps, limitations and a home for iPReS

… Questions

Thank you all… very much
Enjoy the week ahead and everything Austin has to offer.

Find me on Apache lists

lewis.j.mcgibbney@jpl.nasa.gov

lewismc@apache.org

@hectorMcSpector