Enabling Linguistic Analysis of Scientific Metadata through Internationalizing NASA JPL's PODAAC

Lewis J. McGibbney Ph.D[1], Kim D. Whitehall Ph.D[1], Chris A. Mattmann Ph.D[1], Jane R. Wyngaard[1], Ph.D WPhillip Carter[2]
[1] NASA Jet Propulsion Laboratory/California Institute of Technology, Pasadena, CA, United States
[2] School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, United States

Abstract

This paper describes the iPReS project, which provides a web service-based framework for i18n-type (internationalized) access to scientific data products and product metadata contained within the NASA Jet Propulsion Laboratory Physical Oceanography Distributed Active Archive Center, otherwise known as PO.DAAC [0]. PO.DAAC is an element of the Earth Observing System Data and Information System [1], which freely provides science data to the global community of users. The mission of the PO.DAAC is to preserve the quality of NASA's oceanic and climate data and make these universally accessible. iPReS directly shadows this mission objective.

Introduction

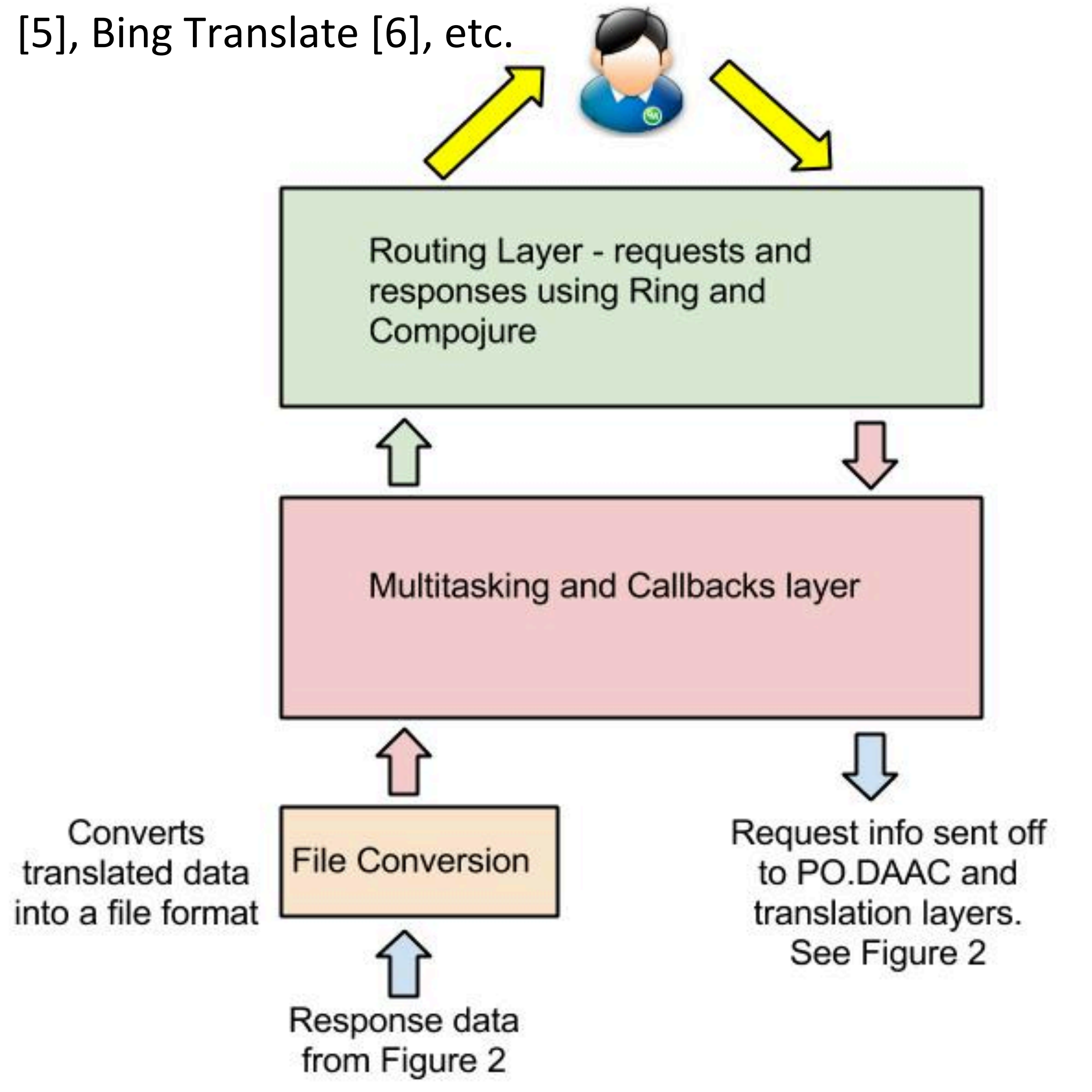
Collaboration and cooperation between various business entities globally is limited by the ability to accurately and efficiently exchange information and communicate across language barriers with limited or no human effort. This challenge also exists within the scientific community, and is compounded by the heavily nested, multi-dimensional (typically array-based) nature of scientific datasets. The nature of such data is a phenomenon that few outside the scientific arena ever encounter. The dimensionality of these data encourages misinterpretation especially when metadata, data documentation and specifications are written in languages unknown to the user.

The iPReS Project is a prototype of a client-facing middleware software component that translates a scientist's query into a query that will search the entire archiving center data irrespective of the linguistic nature of datasets and return the linguistic data from these sources in the queried language. As such, iPReS responds to the aforementioned linguistic hurdles associated with scientific datasets. It further also supports data exploration and discoveries, especially with the emergence of the *Data Scientist*, whose role is to extract knowledge from multiple and some cases "messy" data sources.

Leveraging the Apache Tika Content Analysis Toolkit and Tika Translate API

The **Apache Tika open source content analysis toolkit** "…aims to be the 'digital babel fish', allowing the user to understand any file automatically, rapidly and accurately." [8] Apache Tika was considered a valid candidate for developing a client-facing middleware component for a number of reasons. Demonstrated accuracy and efficiency of translating scientific documents, as well as ease of implementation in web services.

Tika also offers a Java friendly API capable of **automatic media-type detection**, **analysis** and **feature extraction**, and has been used in application scenarios ranging from, Earth science, though biomedicine and business, to search engine indexing, Web content analysis, and patent retrieval (amongst many others). Of particular interest to the iPReS Project is the recent addition of the Tika ***Translate*** **API** which was designed and implemented to provide language translation capabilities to the Tika toolkit. This API provides a number of options for the translation of text from a source to target language(s). Translation options include in house machine translation using existing machine translation software e.g. Joshua [3] and Moses [4], as well as web service enabled tools e.g. Google Translate [5], Bing Translate [6], etc.



Routing Layer - requests and responses using Ring and Compojure

Multitasking and Callbacks layer

Converts translated data into a file format

File Conversion

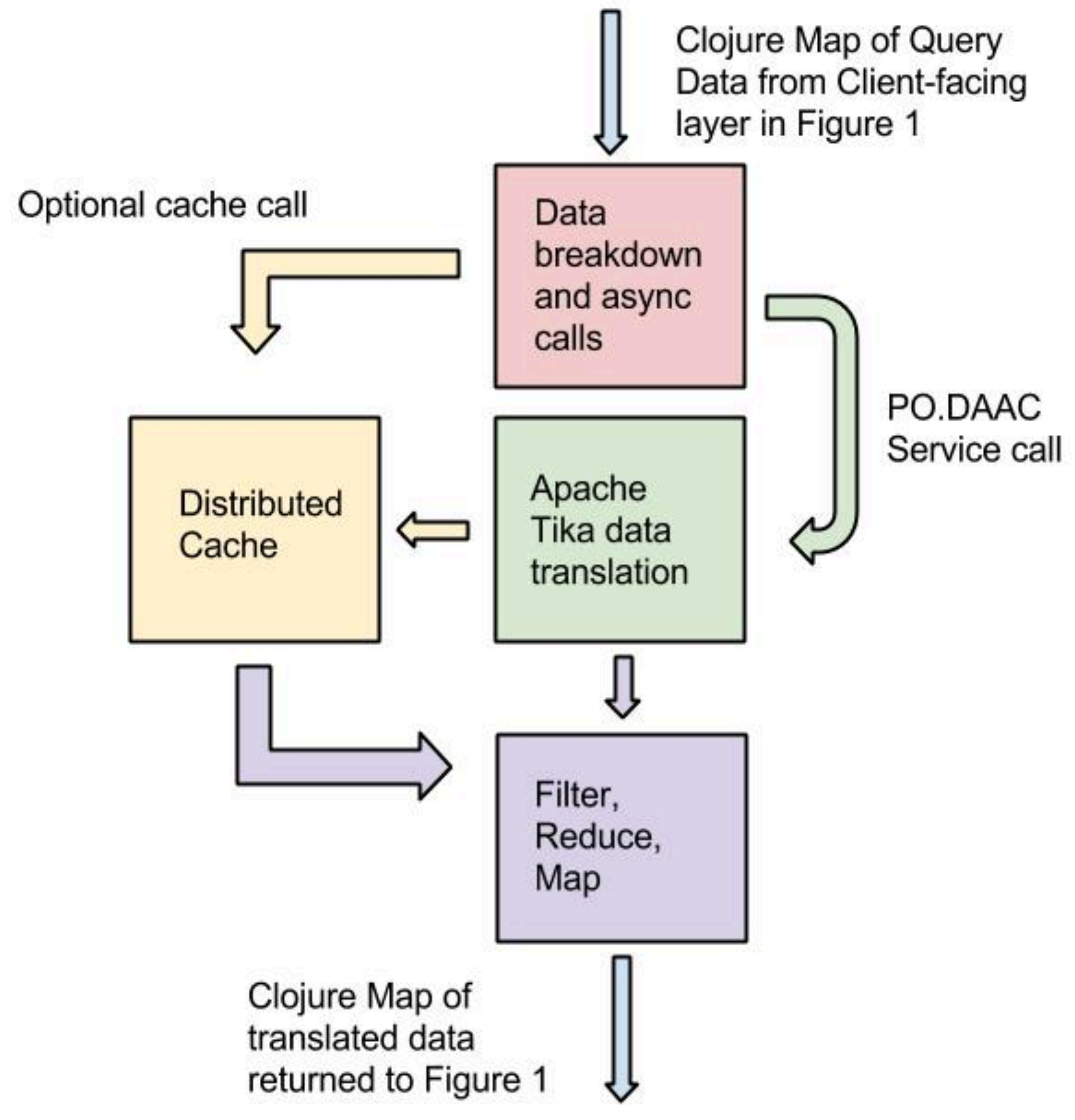
Request info sent off to PO.DAAC and translation layers. See Figure 2

Response data from Figure 2

Figure1: Client facing REST workflow

iPReS Framework Project

A number of components comprise the client-facing service and the server-facing PO.DAAC interfacing Web client



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

Figure2: iPRes PO.DAAC interaction workflow

- **Client Facing REST Workflow**; Figure 1 (which should be interpreted from top to bottom) displays an architectural overview of the client-facing service with which system users interact. This aspect of the framework is ultimately responsible for accepting a request, validating parameters (based upon the PO.DAAC web services REST API specification), and sending the request information to the PO.DAAC-facing end.
- **iPReS PO.DAAC Interaction Workflow**; Figure 2 represents an overview of the PO.DAAC-facing layer responsible for interpreting a request, deconstructing input data into individual query parameters, and making asynchronous calls to both a distributed cache and the PO.DAAC service.

Future work

- Continue to build on the available machine translation language models (language packs) available for both Joshua and Moses.
- Provide command line installation utilities for Machine Translation libraries.
- Undertake detailed analysis of translation accuracy with a focus on improving language translation for scientific metadata.

Conclusion

The iPReS project is aimed at fulfilling PO.DAAC's primary mission of preserving data (in particular ocean and climate data) and making these universally available and meaningful. This is achieved through internationalizing the petabytes of metadata made available to the scientific community using open source technologies (such as Apache Tika), which are licensed under permissive licenses such as the **Apache License v2.0** for the benefit of all. As the iPReS project nears its latter stages, we are pleased to state that it is currently in use within a number of development environments demonstrating the functional capabilities described within this paper. The project will soon enter into a community engagement effort we will further validate and improve this work.

Resources

[0] PODAAC - http://podaac.jpl.nasa.gov/
[1] EOSDIS - https://earthdata.nasa.gov/
[2] Apache Tika – http://tika.apache.org
[3] Joshua - http://joshua-decoder.org/
[4] Moses - http://www.statmt.org/moses/
[5] Google Translate - https://translate.google.com/
[6] Bing Translate - https://www.bing.com/translator/

Acknowledgements

The entire iPReS project effort has been enabled through the CS46x 2014-2015 Capstone joint project effort between NASA JPL and Oregon State University. The authors would like to express appreciation and thanks to a number of individuals from both institutions but in particular the Program Manager D. Kevin McGrath for organizing the program and enabling engaging work of this nature.

Contacts

Please direct any correspondence or feedback to
Dr. Lewis J. McGibbney –
lewis.j.mcgibbney@jpl.nasa.gov