

# Finding Architectural Knowledge in Emails and Issues

Andrew Lalis

October 18, 2021

## **1 Abstract**

This is the abstract!

## Contents

<a href="#">1 Abstract</a>	1
<a href="#">2 Introduction</a>	2
<a href="#">3 Categorization of Architectural Knowledge</a>	2
<a href="#">4 Preparing Data for Analysis</a>	2
<a href="#">4.1 About the ak-tagger Program . . . . .</a>	2

## 2 Introduction

This is a thesis about finding architectural knowledge in emails! Wow!

## 3 Categorization of Architectural Knowledge

According to Tom’s research [1] we know lots about architecture.

## 4 Preparing Data for Analysis

While den Boon’s ArchDetector web application provided a solid foundation for extracting normalized data from various Apache mailing lists (in the form of *mbox* files) and Jira issues[1], it was not designed with the foresight to accommodate further developments, so a new, simple command-line application was developed to improve collaborative workflow of categorizing and exporting data.

This application is called **ak-tagger**, and its source and usage documentation may be viewed online at [this paper’s associated GitHub repository](#). A short description of its functionality will be supplied inline with this paper, but please refer to the online documentation for any further information or to report issues.

### 4.1 About the ak-tagger Program

The architectural knowledge tagger program, colloquially known as **ak-tagger**, functions as a companion to the ArchDetector web application, that provides a more efficient interface for tagging and evaluating the categorization of sources of architectural knowledge, and sharing this information with others. The program is written in the [D programming language](#), and can be compiled to a native executable with any D compiler (DMD, GDC, LDC) that provides the **dub** build tool.

The ak-tagger program works with raw JSON data sets that are grouped into files based on the query that produced the set of results. Thus the program is

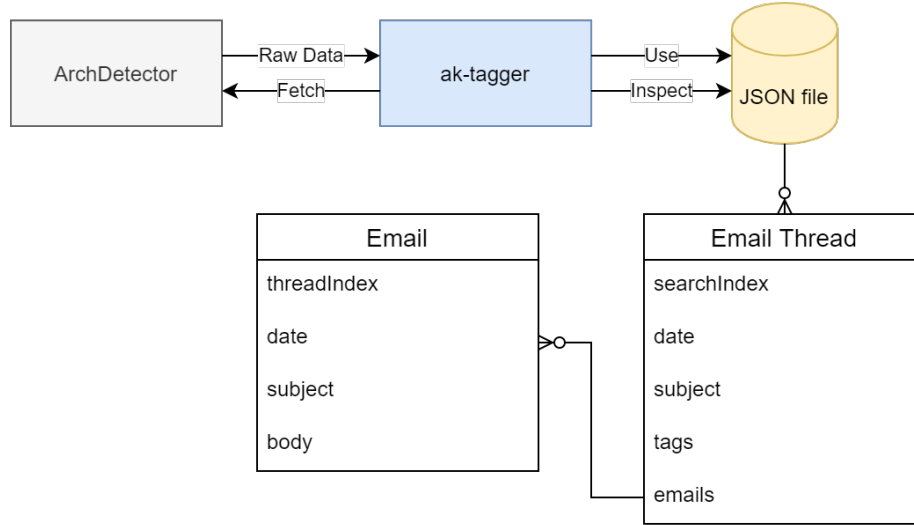


Figure 1: System diagram for the ak-tagger program’s data usage.

designed primarily as a tool for analysis of prepared search queries and their results, as obtained from an instance of the ArchDetector REST API.

There program has several distinct functionalities:

1. **Fetch** - The user is able to fetch a raw dataset from the ArchDetector API and save it to a JSON file for sharing or their own use.
2. **Use** - Fetched datasets may be explored and categorized with a simple interactive command-line interface. Offers utilities such as exporting email threads as formatted text files for easy viewing, and modifying an email thread’s set of tags.
3. **Inspect** - Datasets may be inspected to produce formatted aggregate information that’s easier to use than the entire raw JSON file, including some aggregate information.

The general workflow for a user is to first *fetch* a dataset from ArchDetector, then save it as a JSON file.

## References

- [1] Tom den Boon. Exploring the effectiveness of search engines for finding architectural knowledge in open source repositories. *University of Groningen, Faculty of Science and Engineering Student Theses*. 2021.
- [2] Kruchten P., Lago P., van Vliet H. (2006) Building Up and Reasoning About Architectural Knowledge. In: Hofmeister C., Crnkovic I.,

Reussner R. (eds) Quality of Software Architectures. QoSA 2006. Lecture Notes in Computer Science, vol 4214. Springer, Berlin, Heidelberg.  
[https://doi.org/10.1007/11921998\\_8](https://doi.org/10.1007/11921998_8)