

Project: Speech and Debate Event Timer Android App

Language: Java

Published at: [github.com/PTNobel/SpeechAndDebateTimer](https://github.com/PTNobel/SpeechAndDebateTimer)

Purpose: Speech and Debate Tournaments require presiding officers, individuals overseeing debates, to make specific hand signals at certain times during speeches and to randomly choose speakers. Also, the presiding officer's timer is not allowed to make any noise. To address all of these restrictions, I made an app for Android phones which changes colors to notify the presiding officer when they need to make hand signals. The app also provides a random number table at the bottom of the screen to facilitate fair selection of speakers. I have used it at multiple tournaments during this year.

Project: Music Player Controller (`musicctl`)

Language: Python

Published at: [github.com/PTNobel/musicctl](https://github.com/PTNobel/musicctl)

Purpose: I use three different programs to play music, each with their own programming interface (API). In order to use one set of keyboard shortcuts to control my music, I developed `musicctl` to be an abstraction for the three programs. Please see Figure 1 for an overview of the programs structure.

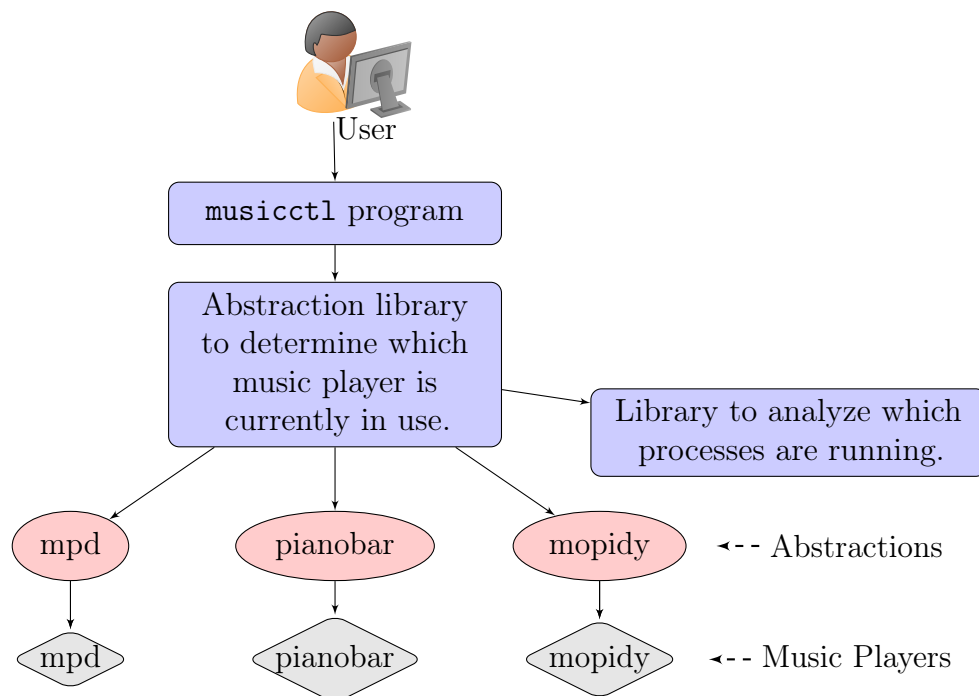


Figure 1: Software Topology for `musicctl`: Boxes denote libraries and programs I wrote, ellipses denote classes I wrote, and diamonds denote software others wrote.

Project: Binary File Modifier

Language: Python

Current Users: Hewlett-Packard

Purpose: My father, a software developer for Hewlett-Packard, mentioned that he needed to a way to modify binary files by inserting a fixed number of bytes at regular intervals in the file. I wrote a program which reads in a binary file, and given the interval, inserts the needed constant data. To facilitate testing the program, I wrote another program that generates test binary files. HP now uses this frequently.

Project: Driving Distance Calculator

Language: Python

Purpose: My high school's Varsity Academic Team Coach was trying to design a proposal for a new division map for our league. The hardest part of designing these maps is ensuring that no school has to travel excessively to a match. Coach Roy was complaining about how difficult the process would be. I researched Google's Maps API, and then implemented a Python script which generates a spreadsheet of driving distances between all the schools in the league. I then pulled the spreadsheet into Excel and used conditional formatting to highlight nearby schools. Coach Roy used the spreadsheet when designing the maps.

Project: Fixing a Flaw in a Existing Open-Source Project (`gitlog.sty`)

Language:  $\text{\LaTeX}$

Published at: [github.com/PTNobel/gitlog](https://github.com/PTNobel/gitlog)

Purpose: When I attempted to use the  $\text{\LaTeX}$  package `gitlog.sty`, I discovered a large limitation. I fixed this limitation by rewriting the relevant portions of `gitlog.sty`. I converted the existing data structures into non-exclusive data structures that could be loaded multiple times. I shared the changes with the developer of `gitlog.sty`. The changes are under review.

Project: Document Renderer (`file_watch`)

Language: Python

Published at: [github.com/PTNobel/file\\_watch](https://github.com/PTNobel/file_watch)

Purpose: In document description languages like  $\text{\LaTeX}$  and Markdown, one writes their document as code and then renders that code to see what the document looks like. When writing essays and papers using  $\text{\LaTeX}$  or Markdown, I often want to see them rendered each time I save the file. To do this, I wrote `file_watch` which renders the document after each save. Please see Figure 2 for an overview of the programs structure.

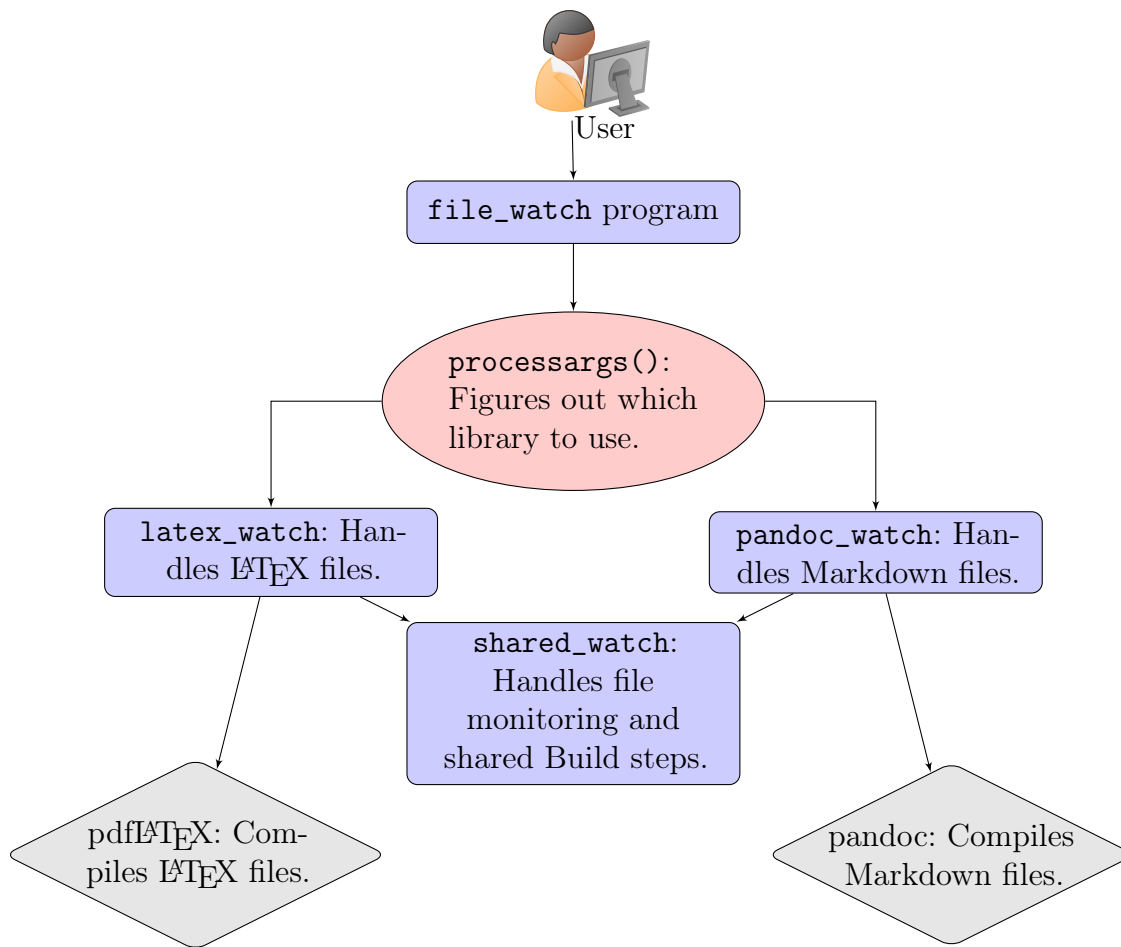


Figure 2: Software Topology for `file_watch`: Boxes denote libraries and programs I wrote, ellipses denote classes I wrote, and diamonds denote software others wrote.