

# Cloudy Music: Streaming Media Through the Cloud

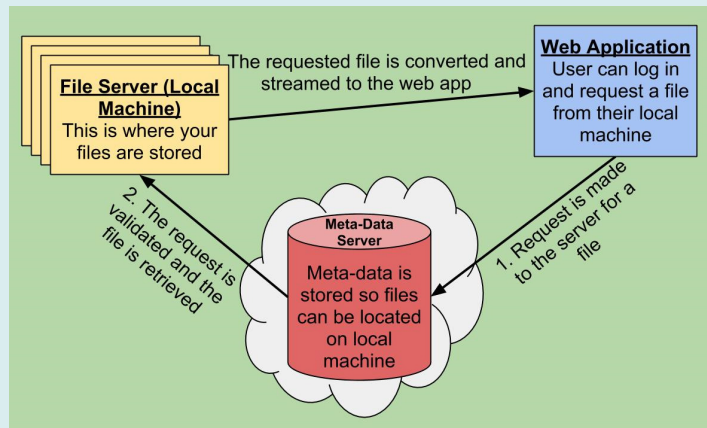
Students: Cameron Bjorklund and Nick Goble ♦ Advisor: Peter Bui  
 Computer Science ♦ University of Wisconsin-Eau Claire



## Problem

There are many great services that provide cloud storage and remote access to stored files and media. To use these services one must upload all of their files to the cloud. Typical residence upload bandwidth is extremely limited, so uploading a large amount of files can take hours.

## Design

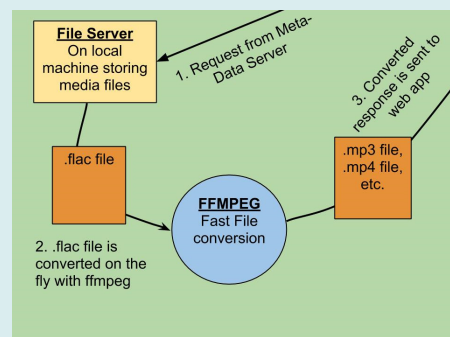


## Solution

- ♦ Only use a small amount of data to store locations of media
- ♦ Only fetch certain files when needed instead of all of them
- ♦ Limit the amount of network traffic
- ♦ Only convert files when necessary, not all at once

## File Conversion

- ♦ Done quickly on the fly when a file is requested by the meta-data server
- ♦ File conversion is done with ffmpeg
- ♦ Can convert from .flac to many different file formats and vice versa



## File Server Functionality

- ♦ Scan for media files within designated directories
- ♦ All media files found locally have their information processed and new information is sent to the meta-data server

## Meta-Data Server Functionality

- ♦ Receives meta-data from file server(s) in the form of JSON and stores that meta-data
- ♦ Processes requests (query, retrieve, etc.) from the web app
- ♦ Determines best file server for streaming, and sends message to begin streaming
- ♦ Tags media based on its meta-data

## Web Application Functionality

- ♦ Display media library to user
- ♦ Sort media by type
- ♦ Request media to download from your file server(s)

## Database Storage Formats

### File Server

Key: File_ID	Pair: A dictionary of key, value pairs wrapped in JSON
<b>JSON</b> <ul style="list-style-type: none"> <li>• MediaTitle</li> <li>• Artist</li> <li>• TimeStamp</li> <li>• Duration</li> </ul>	

### Meta-Data Server

Key: File_ID	Pair: Dictionary
<ul style="list-style-type: none"> <li>• Host</li> <li>• Location</li> <li>• Tag</li> <li>• MediaTitle</li> <li>• Artist</li> <li>• Duration</li> </ul>	

## Conclusion

By using a meta-data server in the cloud as a median, we were able to avoid having to upload an entire media library to a remote server. Instead, limit network traffic by reducing the size of your upload to simple text information. Then use that information to locate files without having to move/copy them. Also we are able to increase efficiency by only converting files when they need to be converted instead of converting everything at once.